Graz University of Technology

Institute for Computer Graphics and Vision

Dissertation

Object detection by partial shape matching, category models and joint segmentation

---

Hayko Riemenschneider

Graz, Austria, 2012

Thesis supervisors

Prof. Dr. Horst Bischof

Prof. Dr. Aleš Leonardis

To my mom.

Deutsche Fassung:
Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008
Genehmigung des Senates am 1.12.2008

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

Graz, am ……. ……..
(Unterschrift)

Englische Fassung:

STATUTORY DECLARATION

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

……. ……..
date (signature)

Abstract

In this thesis three fundamental tasks in computer vision for object detection are addressed: a) shape-based feature extraction, description and matching, b) object category modeling for detection, and c) joining detection, localization and segmentation for improving performance. All of these topics are highly related to each other and novel and faster solutions are provided for the task of object detection.

The goal of this thesis is in partial shape description and matching, category model learning, as well as joining the multiple stages of the object detection into a coherent optimization. The focus is placed on the following underlying challenges. First, feature extraction for shape-based recognition is investigated. The extraction is brought from single pixel evidence to a method integrating mid-level information based on regions. This context integration helps to better distinguish local edge information by considering the underlying region boundaries. Second, considering the essential cue of shape one requires a description, which captures the geometry of the object in a structured and efficient form. This description is built on the sequential information derived from Gestalt laws such as continuity and connectedness. Third, the recognition of shape features requires partial matching. A balance between partiality and similarity is achieved by efficiently evaluating all correspondences and lengths in a 3D similarity tensor for matches between two sequences of points. Fourth, the partial matching of shape is then investigates for rigid template models and learning implicit category models for object detection. Fifth, a method for joining bottom-up processes for semantic detection is investigated. The low-level features are used for classification, detection and segmentation of object instances by combining their evidence from multiple stages into a single optimization.

State-of-the-art results are achieved for edge extraction, shape retrieval, shape-based object detection and localization of overlapping and distorted object instances, each evaluated on respective reference datasets.

Kurzzusammenfassung

In der vorliegenden Arbeit werden drei fundamentale Aufgaben für Objektdetektion in der Bildverarbeitung behandelt: Extraktion und strukturierte Beschreibung von Formen für das Finden von partiellen Ähnlichkeiten, die Modelling von Objektkategorien, sowie das Verbinden von Lokalisierung und Segmentierung für Objektdetektion. Alle diese Themen sind eng miteinander verknüpft und neue und schnellere Lösungen für den Bereich Objektdetektion werden gezeigt.

Das Ziel dieser Arbeit ist die partielle Beschreibung und der Abgleich von Formen sowie das Verbinden mehrere Stufen der Objektdetektion zu einer vereinheitlichten Optimierung. Der Fokus ist daher auf den folgenden zugrunde liegenden Herausforderung: Erstens wird die Extraktion von Merkmalen für die automatische Erkennung untersucht. Die Extraktion basierend auf einzelnen Pixeln wird auf eine Methode erweitert, welche Informationen aus Regionen integriert. Zweitens wird eine Beschreibung von Form vorgestellt, welche die Geometrie in einer strukturierten und effizienten Weise beschreibt. Dies baut auf Sequenzinformationen auf, die durch Gestaltregeln wie Kontinuität abgeleitet werden. Drittens verlangt die Erkennung von Formen einen partiellen Abgleich. Es wird eine Balance zwischen Teillänge und Ähnlichkeit durch einen Abgleich aller Korrespondenzen und Längen erzielt. Viertens, diese partiellen Korrespondenzen werden für Objektdetektion und Kategorie-Lernen gezeigt. Fünftens wird eine Methode zur Verbindung der einzelnen Schritte für eine semantische Detektion untersucht. Die Bildmerkmale werden durch Klassifikation, Detektion und Segmentierung von Objektinstanzen in einer gemeinsamen Optimierung kombiniert.

State-of-the-art Ergebnisse werden in den Bereichen Kantendetektion, Form-Suche sowieso Detektion von Objekten erzielt, was durch Evaluierungen auf Referenzdatensätzen für die jeweiligen Aufgaben gezeigt wird.

So Long, and Thanks for All the Fish.

Douglas Adams

Acknowledgments

Throughout the time of doing my PhD research, I had the luck to experience the guidance, mentoring, friendship, assistance, criticism and love of many amazing people. It is their commitment that brought me to the position with the knowledge and abilities I am today.

First of all, I would like to express my gratitude to my advisor Horst Bischof, who permitted me to pursue a PhD at the Institute for Computer Graphics and Vision. Thank you for giving me the opportunity to experiment with ideas and freely select topics of my choice in the never ending fascination that is computer vision, and for your support and advice during my years at the ICG. I would also like to thank my second advisor, Ales Leonardis, for his comments in the final stages of the thesis.

I was fortunate to work in a group of motivated and inspiring researchers, where colleagues became friends. Hence I would like to express my appreciation to these people. In particular Michael Donoser for his guidance and our endless discussions inside and outside vision. Sabine Sternig whom I don't even know how to compress my gratitude without writing another thesis. But there's Peter Roth who said he who writes a second thesis, didn't do the first one properly. More thanks go to the company and help of Thomas Mauthner, Stefan Kluckner, Peter Kontschieder, Matthias Rüther, and the people participating in the reading groups and tech-talks, which allowed to dive deep into the realms of research.

Additionally I want to thank my friends, who despite my late nights and initial avoidance of sun, tried to keep in touch during the PhD years. Thank you for the inspirational and relaxing FreieFreitage events. Thank you for the Tuesday lunches. Thank you for the traveling. The care-free moments! The Waci way of life. The social activities around the ICG. It was a great pleasure to meet the StripeLord, the chef cook Papa G, the incredible machines builder Christian Reinbacher, the BeerKicker, the endless motivation for TVlessAndHappy living outside of work from Manfred Klopschitz. Andy Wurm's positive attitude and eyes for another side of life. And recently Samuel, Rene and Stefan - the next generation honorarily defending the challenge of the last man standing.

Yet most importantly, I want to thank my family and Kathi. I cannot imagine this work, or anything else, being done without you.

Contents

List of Figures

List of Tables

We do what we must because we can.
For the good of all of us.

Still Alive, Portal

1 Introduction

A major challenge in computer vision is the automatic detection and segmentation of objects in real-life images containing clutter and noise. For humans the recognition of an object and consequent localization is highly interleaved, however often seems like a trivial task. Questions like what is that object and where is it in the world can already be answered by young children.

In computer vision one goal is to teach a computer how to recognize, detect and segment objects in images automatically. Though tremendous progress in this field has been achieved over the years, the tasks are not yet solved. These tasks are difficult due to complexity in real-life images. The images can contain background clutter or have low contrast. Additionally, the object can have a different appearance since its color or texture changed. Further, there can be many overlapping objects, which makes it difficult to distinguish them.

In this thesis we propose several methods for improving the performance and speed for such object detection systems. In particular the following three topics are addressed: a) shape extraction, description and matching, b) object category modeling for detection, and c) joining detection, localization and segmentation for improving performance.

Contents

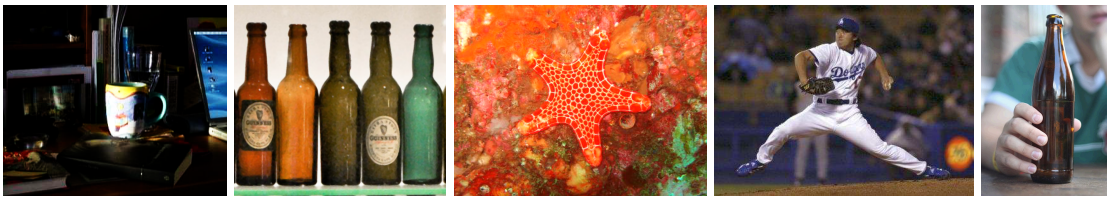
1.1 Motivation

The central goal in this thesis is an automatic and efficient solution for object detection. However, this goal has a number of challenges, which provide many aspects for research.

Images may contain many visual variations and background information confusing the object detection task. These stem from various sources such as low contrast, varying textures, noise from background clutter and deformations due to articulations or occlusions. For humans, it is still possible to recognize the object categories, detect where the objects are, and even draw their exact outlines. For most of the objects as shown in Figure 1.1 this is because there is an underlying cue.

Shape is one of the most promising cues in images, as it describes the outline and layout of objects while providing invariance to a number of these variations in images. For example, it is sufficiently invariant to changes in texture and illumination. However it captures the common properties of instances of the same kind of category. Early research showed that in many cases shape is more generic to categories than color or texture [21]. It is argued that the function of an object is defined by its shape and not its low-level appearance [122].

Further, despite the deformations due to articulation and occlusion of the object, the perception of the shape of objects remains easily understandable - at least for humans. Early research showed that for humans small parts are enough to recognize objects [22]. Even if only a fraction of the object is visible, humans can cope with the partial recognition and understand the visual content. There is another underlying cue, which organizes object parts inherent to an object category. If parts are occluded from the perception, humans also group parts and then infer missing information [104].



Contrast Texture Clutter Articulation Occlusion

Figure 1.1: Complexity of real-world images containing low contrast, varying texture, background clutter, articulation and occlusion while still maintaining the common parts, which is the shape of the objects. Despite this complex nature, for humans there is little difficulty in recognizing and locating the object instances.

1.2 Thesis Goals

The grand goal is to find novel methods for improving the detection performance and speed for large scale applications, for example understanding urban environments captured in massive amounts of street-level imagery. This requires a lot of tolerance to variation and high efficiency in processing. Hence the research in this thesis is concentrated on three topics within object detection.

First, a focus is placed on the powerful underlying cue of shape. It provides a generalization for object categories and an unrivalled efficiency due to the vast amount of data reduction from richly texture images to mere contour responses. For this it is investigated how to extract shape features from images. Further, it is researched how the extracted shape can be efficiently described and matched. The goal is to exploit the underlying ordered structure of shape for better discrimination, partiality and efficiency in runtime.

Second, a focus is set on the generalization of groups of objects into categories. Along this line it is researched how to develop object detection approaches, which can deal with the variations present in images and objects. The goal is to develop partial matching methods, which rely on rigid template models, and learned category models, which implicitly handle the underlying noise and deformations in images.

Third, a focus is placed on semantic detection for improving detection performance. For this it is investigated how to join multiple stages of object detection approaches. Each of the underlying tasks has benefits, which can be exploited in a combined solution. The goal is a scene understanding, where the recognition provides the detection with

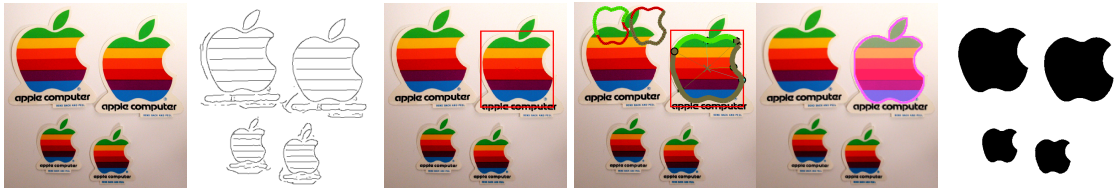


Figure 1.2: Five parts of recognition covered in this thesis: Shape-based feature extraction and structured description, coarse detection up to bounding boxes, finer localization of the local supporting parts, accurate outlines by full object segmentation, and last but not least recognition of multiple instances of objects within an image.

instance evidence and further the localization guides the accurate segmentation of object instances, which in turn loops to the recognition.

Figure 1.2 gives an overview of these goals in this thesis, including feature extraction, detection, localization and segmentation objects. For all these tasks the basis is the extraction of shape-based features, which are described to capture the structure in sequences. Detection is the coarse process of finding where an object is present in an image. Localization of its local support is a fine process to align the feature evidence. Segmentation is the most accurate process exactly outlining all parts of an object. An additional stage not to be underrated is detection of multiple instances, which requires discrimination between objects, when more than one instance is present in the image.

1.3 Outline and Contributions

In the following we briefly outline each of the chapters, the related work in this field, and our contributions introduced in this thesis.

1.3.1 Contours from Region Boundaries

In the second chapter, we look at a novel extraction and perceptual grouping of shape in terms of image contours. Contours are a low-level reduction of information in an image, which describe the changes between the visual content. These changes, causing a contour, have four sources, which are discontinuities in depth, discontinuities in the orientation of the surface, changes in material properties and changes in scene il-

lumination [135]. Related work focuses on small support windows for detecting such underlying sources. As a result this limits the success of perceptual grouping as post-processing leading to fragmentation of contours.

The research in this chapter introduces a novel method for contour extraction by finding the most stable region boundaries [51, 163]. The idea is to move away from analyzing interactions of local pixelwise gradients, and instead integrate mid-level information by analyzing underlying regions that support the local gradient magnitudes. The integration of region support uses a hierarchical data structure denoted as the component tree, which is a unique multi-level representation of the obtained gradients. Contrary to related work we detect stable contours by comparing the shape of regions along this hierarchy. In such a way, we are able to measure the stability for every region boundary and return the most stable ones as our boundary detection result. As our stability analysis uses regions as basic perceptual units, we are bringing region context into contour detection without additional complex perceptual grouping. The main contributions are

- Novel integration of mid-level context cues from regions
- Efficient calculation of shape similarity between regions
- Linked region boundaries without perceptual grouping

1.3.2 Structure in Partial Description

In the third chapter, we look at the description of shape and what characteristics and invariances are important when trying to capture the shape of an object. Related work in this field focuses on global, semi-local or local description, which is often affected by drawbacks. Global descriptions do not cope well with occlusion whereas local descriptions are missing a global structure. Ideally the description is invariant to rigid transformations and articulations. Further it is robust to non-rigid deformation, occlusion and is unaffected by background clutter.

The research in this chapter proposes a novel structured description of shape [49, 50, 53, 108, 164, 165]. This description denoted as *Structural Measurement Descriptor* (SMD) is inspired by the psychological principles of good continuation and collinearity [104]. We focus on the continuity of points and our representation of shape is defined as an ordered sequence of points. Since each point has a unique predecessor and successor, it

allows us to take measurements in a structured fashion. Further, it builds a hierarchical and partial description within a single coherent descriptor. From the general formulation we derive specific instances for shape description of closed regions given by binary masks, arbitrary long image contours and local shape fragments in cluttered images. The main contributions are

- Novel structured description of shape exploiting continuation
- Efficient formulation to include hierarchical and partial shape information
- Specific description instances for regions, contours and fragments

1.3.3 Partial Similarity for Region Matching

In the fourth chapter, we look at the partial similarity for closed region matching. Related work has a long history looking at closed regions for shape retrieval and clustering. For global descriptions there exist distance functions and deformation costs, which however often require known correspondences or assigned parts. The novel shape description *Structural Measurement Descriptor* encodes all parts into one coherent descriptor, where the spatial extent of parts is not known, which makes this approach incompatible with standard matching or detection frameworks.

Research in this chapter introduces a 3D tensor containing all combinations for correspondence, length and similarity [49, 50]. Focusing on matching closed regions given by a binary mask, we propose a novel method for measuring a balanced partial similarity by means of the Salukwadze distance. The main contributions are

- Efficient comparison of matches in a 3D similarity tensor
- Novel partial matching of arbitrary long point sequences
- Partial region similarity measure for shape matching and clustering

1.3.4 Partial Contour Matching for Object Detection

In the fifth chapter, we look into the partial matching of cluttered image contours for object detection. Related work in this field either relies on piecewise contour approximations, requires meaningful supervised decomposition, or matches coarse shape-based descriptions at local interest points. This has drawbacks due to the error-prone preprocessing steps for edge grouping and loss of structure for independent interest points.

Research in this chapter proposes a novel method for object detection by partially matching image contours to a single hand-drawn rigid shape template [164]. Our method uses all obtained contours for partial matching. The matched contours are efficiently summarized to long salient matches and aggregated by a star-model to form location hypotheses. The efficiency and accuracy of our edge contour based voting step yields high quality hypotheses in low computation time. The main contributions are

- Novel true partial matching of cluttered image contours
- Efficient extraction and grouping of partial matches for center voting
- Partial contour-based object detection by a single rigid template model

1.3.5 Discriminative Fragments for Object Detection

In the sixth chapter, we look into the graphical representations and approaches when learning and evaluating object category models. Related work incorporates learning techniques only for object model generation or for verification after detection. The actual object detection is often performed by low-level chamfer matching [153, 183], which is sensitive to noise and clutter.

Research in this chapter introduces a novel method for object category localization by jointly learning discriminative contour fragments and also the model for the object category [108]. In the learning phase, we interrelate local shape fragments of the object contour with their corresponding spatial location. The shape description is based on the *Structural Measurement Descriptor* and is well-suited for discriminative learning in a random forest learning scheme. For detection we hypothesize object locations and the back-projection of the category model alongside the shape fragments allows to delineate the object outline. The main contributions are

- Novel formulation of shape fragments and category models
- Discriminative fragments and center votes for joint learning and evaluation
- Efficient non-rigid object detection and outlining by actual image evidence

1.3.6 Joining Classification, Localization and Segmentation

In the seventh chapter, we look into the joining of classification, localization and segmentation for object detection. Related work in this field exists in the segmentation of

known objects, complex interaction learning of instances, or full-blown scene understanding goal. The drawbacks of high computational requirements and known object location hypotheses limit the general use of these approaches.

Research in this chapter proposes a novel method which jointly optimizes and implicitly provides detection hypotheses and corresponding segmentations [166]. It is attached to any available generalized Hough voting method. We describe our method denoted as *Hough Regions* formulating the problem as a two-layer graph of the image space and the Hough space, which are connected through the object center votes. This exploits classifier responses, object center votes and low-level cues like color consistency, which are solved by a global energy formulation. For object detection we obtain a pixel-wise assignment for each detected object instance. The main contributions are

- Novel joint formulation of classification, localization and segmentation
- Two-layer image and Hough graph for enumeration of object instances
- Increased tolerance for overlapping instances, changes in scale and aspect ratio

1.4 Applications

The methods developed in this thesis have many applications. They are used in large-scale tasks such as city-scale window and other object detection, which is feasible due to the focus on high efficiency and generalization to other object categories.

The grand goal of object detection is realized in four implementations for this task. A chamfer matching framework based on the novel stable boundaries is a very efficient shape-based detection method. A partial contour matching framework allows clutter-independent and articulation-invariant detection of objects given a single hand-drawn reference template. A category model learning and evaluation framework provides a balance between high efficiency and discrimination power for shape-based object detection. Finally, the joint localization and segmentation framework using rich appearance produces state-of-the-art results, especially for detection of overlapping and articulated objects at reduced runtime.

Further, the partial shape description and matching have been used for object tracking [52], video segmentation [53] and a sketch-based content retrieval for image search [165].

The important thing in science is not so much to obtain new facts as to discover new ways of thinking about them.

Sir William Bragg

2

Contours from Region Boundaries

Many shape-based category recognition methods require stable, connected and labeled contours as input. This chapter introduces a novel method to find the most stable region contours in grayscale images for this purpose. In Section 2.2 we review common edge detection algorithms, of which most only analyze local discontinuities in image brightness, where in contrast our method integrates mid-level information by analyzing regions that support the local gradient magnitudes. In Section 2.3 we describe our method using a component tree where every node contains a single connected region obtained from thresholding the gradient magnitude image. Edges in the tree are defined by an inclusion relationship between nested regions in different levels of the tree. Region boundaries which are similar in shape (i. e. have a low chamfer distance) across several levels of the tree are included in the final result. Since the component tree can be calculated in quasi-linear time and chamfer matching between nodes in the component tree is reduced to analysis of the distance transformation, results are obtained in an efficient manner. The novel detection algorithm labels all identified edges during calculation, thus avoiding the cumbersome post-processing of connecting and labeling edge responses. In Section 2.4 we evaluate our method on two datasets and demonstrate improved performance for shape prototype based localization of objects in images.

Contents

2.1 Introduction

Despite the tremendous progress in the of contour extraction, it is still not clear what the main features are that help to group local evidence into objects and further into categories. Early research showed that in many cases shape is often more generic to categories than color or texture [21]. Hence different authors like [63, 122, 153, 182] proposed methods outperforming appearance-based approaches for several object categories by relying on shape features. All these methods implicitly require stable, connected and labeled contours in the image as underlying representation. This has the purpose of reducing the data amount while retaining the important information about the image content. Such contours are a low-level reduction of information in an image, which describe the changes between the visual content. These changes causing a contour have four sources, which are discontinuities in depth, discontinuities in the orientation of the surface, changes in material properties and changes in scene illumination [135]. Most approaches however rely on the post-processed results of standard pixelwise gradient analysis [35] or learned boundary responses [136, 137]. These methods are considered as state of the art in this field, yet only consider local cues and provide pixelwise results. Post-processing is hence necessary because most detection frameworks require labeled lists of connected edges in images. Perceptual grouping obtains these by analyzing T-junctions [35, 109], multi-branch contour groups and networks [66], or eigenvalue analysis of the contour grouping graph [219].

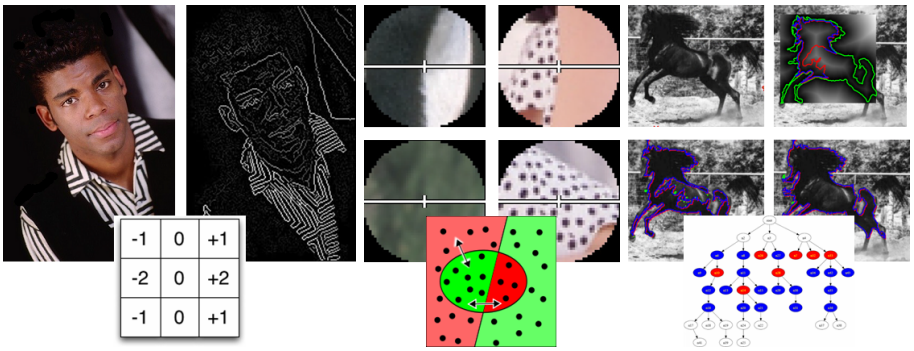
In this chapter we introduce a novel boundary detection method, designed for the application of object detection, which extends purely local edge detectors by additionally analyzing mid-level cues, i. e. regions that support the local gradient magnitude are analyzed to extract the most stable edges. It is this mid-level context information that

determines the stability of boundaries. The extraction is based on analyzing a hierarchical data structure denoted as component tree where connected regions, which are separated by high gradient magnitudes along their boundaries, are linked into a tree structure. Finding the most stable nodes in the tree, by considering shape similarity of the region contours, removes noise and clutter and preserves the important contours for detection. Furthermore, our method automatically labels all obtained contours during calculation. Therefore, no further post-processing is required and the results can be directly passed to any of the available shape-based object localization methods.

The outline of this chapter is as follows. Section 2.2 gives an overview on related work concerning contour detectors and outlines similarities of our method to recent state of the art. Section 2.3 describes our novel boundary detection method in detail. In Section 2.4 we demonstrate on two well-known object recognition datasets that the proposed method reduces clutter and maintains the most important contour responses. We furthermore outline a potential application in the area of object localization by using our boundary responses in an oriented chamfer matching step. Finally, Section 2.5 discusses our results and gives an outlook on future work in this area.

2.2 Related Work

In general, contour detection is one of the most intensively studied problems in computer vision and has many other applications like stereo matching, segmentation, 3D reconstruction or tracking. We follow the distinction between edges and boundaries outlined in [137]. Edges are defined abrupt changes in a low-level image feature like brightness or color gradients, whereas boundaries are connected pixels in the image, which represent a change in pixel ownership between two neighboring objects or entities. The approaches in this field hence can be divided into methods, which simply analyze local intensity differences and the recently popular group of learning based methods, which focus on learning the appearance of boundaries, e. g. for specific tasks [48], or in a more general way for natural images [107, 137]. Of course, learning-based boundary detection methods provide improved results making use of training images. However, these methods have the main drawback of high computation time, for example in the range of twelve seconds [48] and 90 seconds [216, 217]. Even a highly optimized implementation for GPUs using 30 cores still requires about two seconds per image [36]. Furthermore, they require ground truth to learn application specific detectors. Therefore, local edge detectors are still of high interest due to their computational simplicity and speed.



(a) Canny (b) Berkeley (c) Stabs

Figure 2.1: Contour extraction by a) Canny edge detection [35] using a 3×3 Sobel kernel and subsequent linking, b) boundary probability estimation by supervised learning of image features in a local support window [137], and c) our proposed method analyzing the hierarchy of underlying regions for boundary similarity.

Figure 2.1 gives an overview of the most commonly used contour detection approaches. One example is Canny [35] because of its simplicity, effectiveness and high efficiency. Canny detects edges in the image by several consecutive steps. First, Gaussian blurring is applied to reduce the noise of individual pixels. Second, 2D Gaussian derivative responses are calculated to detect pixels of sharp intensity changes by the magnitude and the orientation of the gradient response. The gradient image undergoes a non-maxima suppression to filter out weaker responses. Finally, a connected component analysis based on a hysteresis thresholding heuristic returns the set of final edges.

In [75] it is shown that already very local figure-ground cues provide a better contour support than individual pixel evidence. The local cues, which are derived from area ratios, mass centers and convexity, are combined using a logistic function. The Berkeley edge detector [137] is a supervised algorithm specifically tuned to natural scenes. It is based on analyzing several image features like raw and oriented energy and texture gradients. These image features are then combined to improve boundary detection in natural images. For feature combination, supervised learning is used which exploits a ground truth dataset built from multiple human segmentations, where humans were asked to draw boundaries of equally important things in the training images. This ground truth is used in a logistic regression model to learn an optimal feature combination for natural images. Further work models the image as a fully connected graph and derives a multi-scale probability (mPb) and a saliency of boundaries (sPb). Weights

for these two are then learned for a linear combination of a global probability of boundaries (gPb) [136], which is considered state of the art in the field.

More specific to individual categories is the work for boosted edge learning (BEL) [48]. For BEL the local features are combined with category labels and specific edge responses are learned for a specific task. In [94] contour detection is lifted to recovering occlusion boundaries. The depth discontinuities of an image are estimated from local gradient and 3D depth cues together. Along this research in [191], the object boundaries are extracted by learning motion cues from short sequences, but still only from local pixel evidence.

Our method aims to incorporate larger context than pixels and local support windows inspired by [216, 217]. For this we analyze the underlying regions and exploit a hierarchical data structure denoted as component tree to obtain our stable boundary detection results. Building and analyzing such a hierarchy of nested regions for edge detection was proposed by Najman and Schmitt [149] for image segmentation.

In [149] the dynamics of watershed contours were used to define edges. The method is based on identifying watersheds by a standard flooding process. Starting from local minima in the gradient magnitudes, the image is flooded and when regions merge, the watershed height defines the saliency of the local contour. In contrast to our approach, contour saliency is defined as the minimal intensity contrast between two merged regions along their common boundary. Any stability analysis of the contour is neglected.

Most similar to our approach is the boundary extraction method proposed by Arbelaez [7]. Arbelaez et al. also builds a hierarchical data structure using a stratification index and outlines its equivalence to an ultra-metric distance between regions [8, 9]. This data structure is denoted as an ultrametric contour map, where each level provides a segmentation of the image at various levels of detail. Contrary to our approach, color information is exploited in terms of a contrast function and a color uniformity measure per segment. Furthermore an image pre-segmentation by means of a Voronoi tessellation is required, which increases computational complexity. Arbelaez [7] further exploits the concept of strong causality to provide a saliency value for each segment contour. This in turn allows to extract boundaries of different strengths.

Our work determines the boundaries by analyzing their similarity in shape. This provides a strong cue for consistent regions and hence stable boundaries. An underlying region implicitly encodes the visual changes caused by discontinuities in depth, discontinuities in the orientation of the surface, and changes in material properties.

2.3 Region-based Contour Detection

Our novel boundary detection and labeling method uses regions as basic perceptual units. The context of an underlying region is what gives stability to a contour. In this chapter we propose a method, which uses - similar to most local edge detection methods - a fast standard gradient pre-processing as outlined in Section 2.3.1 and then integrates mid-level context from regions to provide stable clutter-free contour detection. The integration of region support uses a hierarchical data structure denoted as the component tree, which is a unique multi-level representation of the obtained gradients. The component tree and its efficient calculation is presented in Section 2.3.2. The most important part for detecting stable contours is our analysis of the component tree by comparing the shape of regions at different levels. In such a way, we are able to measure the stability for every region boundary and return the most stable ones as our boundary detection result. This stability analysis is defined in Section 2.3.3 and provides the key insight to bringing region context into contour detection without additional perceptual grouping.

2.3.1 Pre-processing

As a first step we obtain a gradient magnitude image, which is used as input to the component tree analysis outlined in Section 2.3.2. Please note, that we can use any gradient map for our stable boundary detection method, for example the gradient responses obtained by the Berkeley detector [137]. But due to its simplicity and much lower computational complexity, we only use simple Gaussian derivatives.

In general image derivatives emphasize high frequencies and thus amplify noise. Therefore, it is required to smooth the image with a low-pass filter prior to gradient calculation using a circularly symmetric filter, since contour responses should be independent of orientation. We use the same sequence of pre-processing steps as the Canny edge detection approach [35]. We first convolve the image with a 2D Gaussian to remove noise and then apply a first order 2D Gaussian derivative filter. Since Gaussian filtering is separable we can use two 1D convolutions in order to achieve the effect of convolving with a 2D Gaussian. As a result we obtain gradient magnitudes and orientations per pixel. In this chapter we neglect orientation information in the following steps and simply use the obtained gradient magnitude map I. The component tree as described in the next section requires pixel values coming from a totally ordered set as input. Therefore, we further normalize the magnitudes and quantize them to an integer range.

2.3.2 Component tree

For detection and labeling of stable boundaries in an image, we build a unique data structure denoted as component tree for the obtained gradient magnitude image. It was originally introduced in statistics [92, 206] for classification and clustering and was redefined by Jones [100] for the field of image analysis as a *"representation of a gray-level image that contains information about each image component and the links that exist between components at sequential gray-levels in the image"*. It is this inclusion relationship, which we exploit to access the underlying regions in an image. Our method compares the stability of the shape of regions across various gradient strengths, which is possible via the nodes in the tree hierarchy.

The component tree can be built for any vertex-weighted graph defined by a triplet $G = (V, E, I)$. V is a finite set of vertices, i. e. the set of pixels from our image with V being a subset of $\mathbb{Z}^2$. Therefore a vertex $x \in V$ in our case is defined by its two coordinates (x_1, x_2). E is the set of edges defining the neighborhood relation of the graph, where Γ is a map from a $x \in V$ to a subset of V, such that for every pixel $x \in V$, $\Gamma(x) = \{y \in V \mid (x, y) \in E\}$. If a pixel $y \in \Gamma(x)$, we say that y is a neighbor of x. In our image setting, the neighborhood relation is defined by the standard 4-pixel neighborhood as

$$E = \{(x, y) \in V \times V, |x_1 - y_1| + |x_2 - y_2| = 1\} \tag{2.1}$$

where I is a map from V to D, and D is any finite set allowing to order all points, e. g. a finite subset of integers. In our case, $D = \{0, 1, \ldots, 255\}$ and the mapping I is defined by the normalized gradient magnitudes that are scaled up to an unsigned 8-bit integer range. The magnitude values I are inverted so that a low value represents a high gradient value. A cross-section of the vertex weighted graph at a level t is defined as

$$I_t = \{x \in V \mid I(x) \geq t\} \tag{2.2}$$

where each possible cross section I_t is a standard non-weighted graph defined by a tuple $G_t = (V, E)$, where E is a binary relation on V being anti-reflexive $(x, x) \notin E$ and symmetric $(x, y) \in E \Leftrightarrow (y, x) \in E$. E for a cross section I_t connects neighboring pixels (x, y) only if $I(x) \geq t$ and $I(y) \geq t$.

We further define linkage between two vertices x and y, if there is a path $P = (p_0, p_1, \ldots, p_N)$ with $p_i \in V$ between x and y so that for every pixel along the path

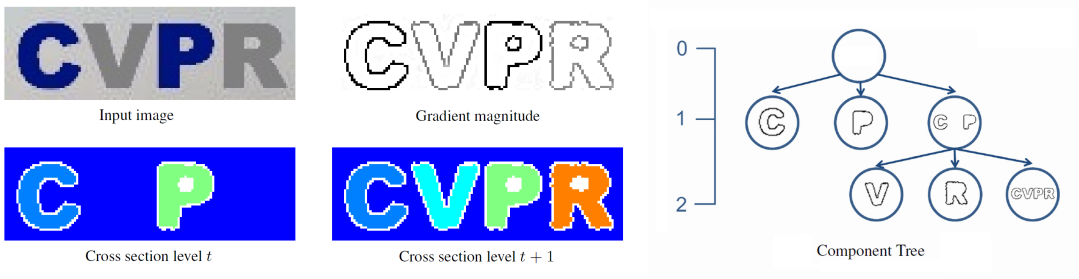


Figure 2.2: Illustration of component tree representation. First row on the left shows the input image and the obtained gradient magnitude response (dark values equal high gradients). Second row shows label results after thresholding the magnitude image at two different levels. Regions correspond to connected areas separated by high gradient values along their boundaries. Cross-sections are related by an inclusion relationship, i. e. regions can only become smaller. This region nesting defines the hierarchical data structure denoted as component tree, as shown on the right. Outer boundaries of these regions are analyzed concerning shape similarity over the levels of the component tree to detect stable ones.

$(p_i, p_{i+1}) \in E$ is valid. We denote a subset $X \subseteq V$ as connected, if any $x, y \in X$ are linked. A connected component C is defined as a maximal subset $C \subseteq V$, where it is not possible to add another pixel $z \in V$ which is linked to any $x \in C$ of the component.

These definitions now allow to define the component tree. We first find the minimum level $t_{min} = min\{I(x) \mid x \in V\}$ and the maximum level $t_{max} = max\{I(x) \mid x \in V\}$. We then calculate all cross-sections I_t of the vertex-weighted graph in the range of t_{min} to t_{max} and find in each cross-section I_t the connected components C^t as defined above to build the component tree structure from the top to the bottom. The obtained connected components C^t can be organized in a hierarchical data structure denoted as component tree because they are nested and their inclusion relationship is defined by $\forall x \in C^t, \exists y \in C^{t+1} : x = y$. Therefore, region size can only decrease from the top to the bottom of the component tree and the root node contains all components merged into a complete representation of the input image. Please note, that in contrast to the hierarchical structure in [7] the localization of the contours is not preserved through different levels of the tree as single pixels are included each level instead of a superpixels.

The component tree can be implemented in many efficient ways [99, 100, 147, 148, 169]. We use the algorithm proposed by Najman and Couprie [42] since it has the lowest computational complexity. Their algorithm runs in quasi-linear time, where the term *quasi* is an amortization of the costs required for the union-find problem. Thus, the worst-case complexity is $O(N \times \alpha(N))$ with $\alpha(N)$ being the inverse Ackermann

function [1], i. e. practically linear. On a side note, the component tree is also the basis for the calculation of Maximally Stable Extremal Regions (MSER) [139]. Recently Nistér and Stewénius [150] even showed that calculating MSERs is possible in linear time returning an ellipse instead of a pixel representation for each region.

In summary, the component tree is a unique multi-level representation encoding the gradient magnitude structure of the input image as illustrated in Figure 2.2. As can be seen regions within the cross-sections of the component tree correspond to connected areas in the image separated by high gradient magnitude values along their boundaries, related to the concept of watershed segmentation. Every node of the tree contains data about the corresponding outer region boundaries, the tree level and the hierarchical relations to other component tree nodes. The shape of some regions stays the same over several levels of the tree, which is the criterion that we use to detect stable boundaries as it is outlined in the next section. Furthermore, since we only want to detect stable boundaries, which are supported by an underlying region, we exclude all regions below a fixed threshold Ω for the region area size.

2.3.3 Stable Region Boundaries

We now use the component tree structure, built as described in Section 2.3.2, for detecting and immediately labeling boundaries in the input image. The underlying idea is similar to the concept of Maximally Stable Extremal Region (MSER) detection as proposed by Matas et al. [139]. MSER detection builds the component tree directly on the intensities of the image and returns the most stable nodes (extremal regions) of the tree as interest region detection results. The stability is estimated by comparing region sizes between nodes in adjacent levels of the tree. Thus, a region is considered stable when its statistical area size does not change abruptly.

We follow a similar principle but focus on analysis of the boundary shape stability between the regions. In contrast to MSER detection, which compares region sizes, we measure how similar the actual shape of the regions is, i. e. if parts of the region boundary remain more or less the same over several levels of the component tree. We define the stability value $\Psi(C_i^t)$ of a connected component C_i^t at a component tree cross-section I_t at level t by comparing its shape to a region $C_j^{t-\Delta}$, which is the connected component linked to C_i^t by moving along the hierarchical structure up to a level $t - \Delta$ (see Figure 2.2), where Δ is a stability parameter of the method. Increasing the value Δ yields

stricter stability constraints, i. e. region boundaries have to be stable over more levels of the tree, which leads to a reduced number of detected boundaries.

To measure the shape similarity between C_i^t and $C_j^{t-\Delta}$ and to identify similar contour parts between the regions, we apply a simplified version of chamfer matching. Since regions within the tree can only grow from level t to level $t - \Delta$ and always are fixed at the same location, chamfer matching is reduced to an analysis of the distance transformation values. In this work we employ a standard chamfer matching, but please note that more sophisticated methods, for example methods additionally considering orientation [183] or in general any partial shape matching method, can be integrated.

Let $\mathcal{B}_i$ and $\mathcal{B}_j$ be the sequence of outer boundary pixels for the current region C_i^t and its linked neighbor $C_j^{t-\Delta}$. As a first step we calculate the distance transformation DT_i on the binary image solely containing the boundary pixels of $\mathcal{B}_i$, which assigns to each pixel the minimum distance to a boundary point. This distance transformation DT_i enables to find partial matches between the boundaries $\mathcal{C}_i$ and $\mathcal{C}_j$ by finding connected boundary fragments $\overline{\mathcal{C}}_j \subseteq \mathcal{C}_j$ fulfilling

$$\overline{\mathcal{C}}_j \subseteq \mathcal{C}_j \rightarrow \forall x \in \overline{\mathcal{C}}_j \, : \, DT_i\,(x) \leq \Phi, \tag{2.3}$$

where Φ is a maximum boundary distance parameter. Finally, for the region C_i^t we set the corresponding stability value $\Psi(C_i^t)$ to the average chamfer distance of the matched boundary pixels $\overline{\mathcal{C}}_j$ by

$$\Psi(C_i^t) \; = \frac{1}{N} \sum_{n=1}^{N} DT_i\,(x_n) \;\; \textit{where} \;\; x_n \in \overline{\mathcal{C}}_j \tag{2.4}$$

and N is the number of matched pixel. In such a way we get a stability score $\Psi(C_i^t)$ and matched connected boundary fragments for every region in an efficient manner, since we only have to look up corresponding distance transformation values.

After calculating the stability values $\Psi(C_i^t)$ for every node in the component tree, we detect the most stable nodes along each path in the component tree similar to [139] and return the corresponding matched boundary parts as detection result. We further assign the level in the component tree in which a boundary is detected as its saliency value. This gives a measurement of the importance of the boundary, since detection at a level t means that all gradient magnitude values of the boundary have to be higher or equal than t. Furthermore, since every detected boundary is connected to a specific node in the component tree, a unique boundary ID can be assigned on-the-fly during component

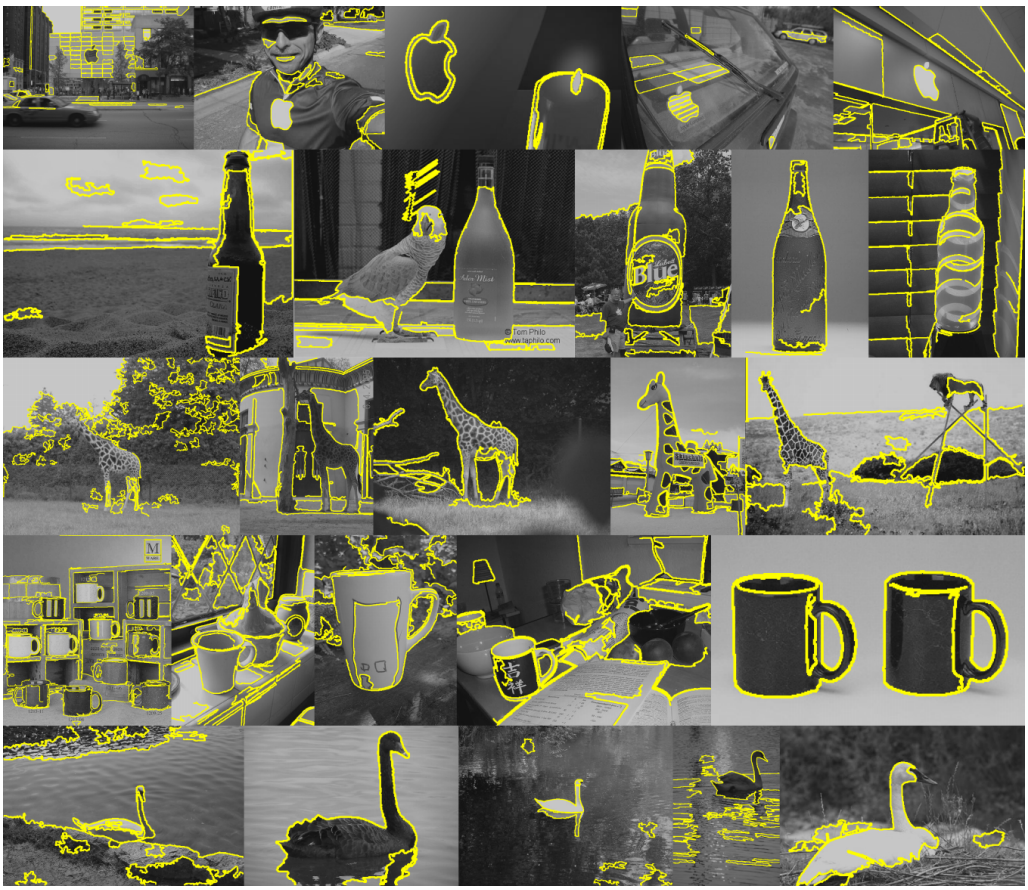


Figure 2.3: Contour detection performance of our proposed stable region boundary extraction method on the five classes of the ETHZ Shape dataset [66].

tree analysis. Therefore, cumbersome post-processing to link and clean contours is not required and the final output of our method is a labeled set of connected boundaries.

Please note, that boundaries detection results as shown in Figure 2.3 are quite different from simply returning the region contours of MSER detection results. First of all, we use the gradient magnitude image instead of the intensity image as underlying representation. Second, we have a different stability criterion analyzing the stability of the shape of the region contours instead of region size stability. Finally, since we identify parts of the region contours that are similar, the returned boundaries need not be closed.

2.4 Experimental Evaluation

The focus of experimental evaluation lies on demonstrating the reduced noise and high retrieval of valuable stable contours. In Section 2.4.1 we use the Berkeley benchmark [138] for experimental evaluation on two well-known object detection datasets. Since our proposed method returns well-connected and stable boundaries, it is well-suited for both partial shape matching and template comparison methods. For binary template search a mask is shifted over the image and similarities to the provided template are calculated. Therefore, in Section 2.4.2 we use our boundaries in an oriented chamfer matching method and illustrate results in comparison to Canny and Berkeley contour detection methods.

Our method has four parameters, which are fixed for all experiments. The minimum considered region size Ω is 400. The stability parameter Δ is 5 and the shape similarity parameter Φ is 10. Furthermore, after obtaining the results, we remove all boundaries with a length below 70 pixels, which is easily done because each boundary is already linked and can be directly accessed by its unique ID.

2.4.1 Object Boundary Evaluation

As a first experiment we adopt the framework of the Berkeley segmentation benchmark [138], which allows us to measure the overlap of detected contours compared to human ground truth data. We evaluate on two well-known object detection datasets, namely the ETHZ Shape classes [66] and the Weizmann Horses [29]. However, instead of human drawn annotation as ground truth, we use the detection annotation for both datasets as a set of figure/ground segmentations. For the ETHZ dataset, we created a new figure/ground segmentation, whereas for the Weizmann Horses they are already provided. In both cases the segmentations highlight the outline of the objects that should be detected in the test image. For evaluation of the performance respective to the defined object-centric ground truth data, different contour detection method are compared to the ground truth in the same manner as specified in the Berkeley segmentation benchmark.

We use precision and recall to measure the quality of the obtained contour responses. Precision in our case is the probability that a detected contour pixel is a true contour pixel. Recall is the probability that a true contour pixel is detected by the algorithm. We calculate the F-measure as a final comparison value defined as weighted harmonic mean of precision and recall. The Berkeley benchmark provides exactly these precision

Algorithm	Canny			Berkeley			Our detector		
Class	P	R	F	P	R	F	P	R	F
ETHZ applelogos	0.02	0.99	0.05	0.08	0.95	0.15	0.12	0.90	**0.21**
ETHZ bottles	0.06	0.99	0.11	0.16	0.95	0.28	0.17	0.84	**0.29**
ETHZ giraffes	0.10	0.99	0.10	0.20	0.90	**0.32**	0.16	0.69	0.26
ETHZ mugs	0.08	0.98	0.15	0.19	0.94	**0.32**	0.18	0.86	0.30
ETHZ swans	0.05	0.98	0.10	0.15	0.94	0.27	0.24	0.82	**0.38**
Weizmann horses	0.14	0.94	0.25	0.18	0.94	0.30	0.33	0.53	**0.41**
Average	0.08	0.98	0.13	0.16	0.94	0.27	0.20	0.77	**0.31**

Table 2.1: Comparison of the overall precision, recall and F-measure performance of each algorithm for the two datasets ETHZ Shape classes [66] and Weizmann Horses [29]. Please note, that the benchmark is very strict given that only true boundaries of objects-of-interest are marked in the ground truth segmentation (resulting in a low precision value). Our boundray detector provides an encouraging improvement over the Canny (18%) and the learned Berkeley edge responses (4%).

and recall curves for each threshold of the image. Since object recognition systems often require binary decisions as input where contour are located, we thus evaluate the pure binary response performance. We select all contours with a gradient magnitude above zero and retrieve a single F-measure per test image. By analyzing binary responses containing all contours, we can focus on the performance benefit for a consequent object detection approach.

Table 2.1 summarizes results of this experiment, outlining the calculated precision, recall and F-measures. We compare results for the ETHZ Shape classes and the Weizmann Horses obtained by the Canny edge detector using default parameters, by the Berkeley edge detector and by our proposed method. The improvements achieved by our stable boundary detector are encouraging for both datasets, yielding an average F-measure improvement over all analyzed classes of 18% compared to standard Canny and of 4% compared to the learned Berkeley detector. It thus matches the quality of the detection results of a supervised method and the speed of a standard Canny method.

The improvements achieved by our boundary detector in these experiments demonstrate how contours become stable when they exhibit region support. The applelogo, bottle, swan and horse classes all contain strong region support for its boundaries. On the other hand, giraffes due to their strong texture deliver less stable region support, which in turn results in a lower recall for our obtained boundaries.

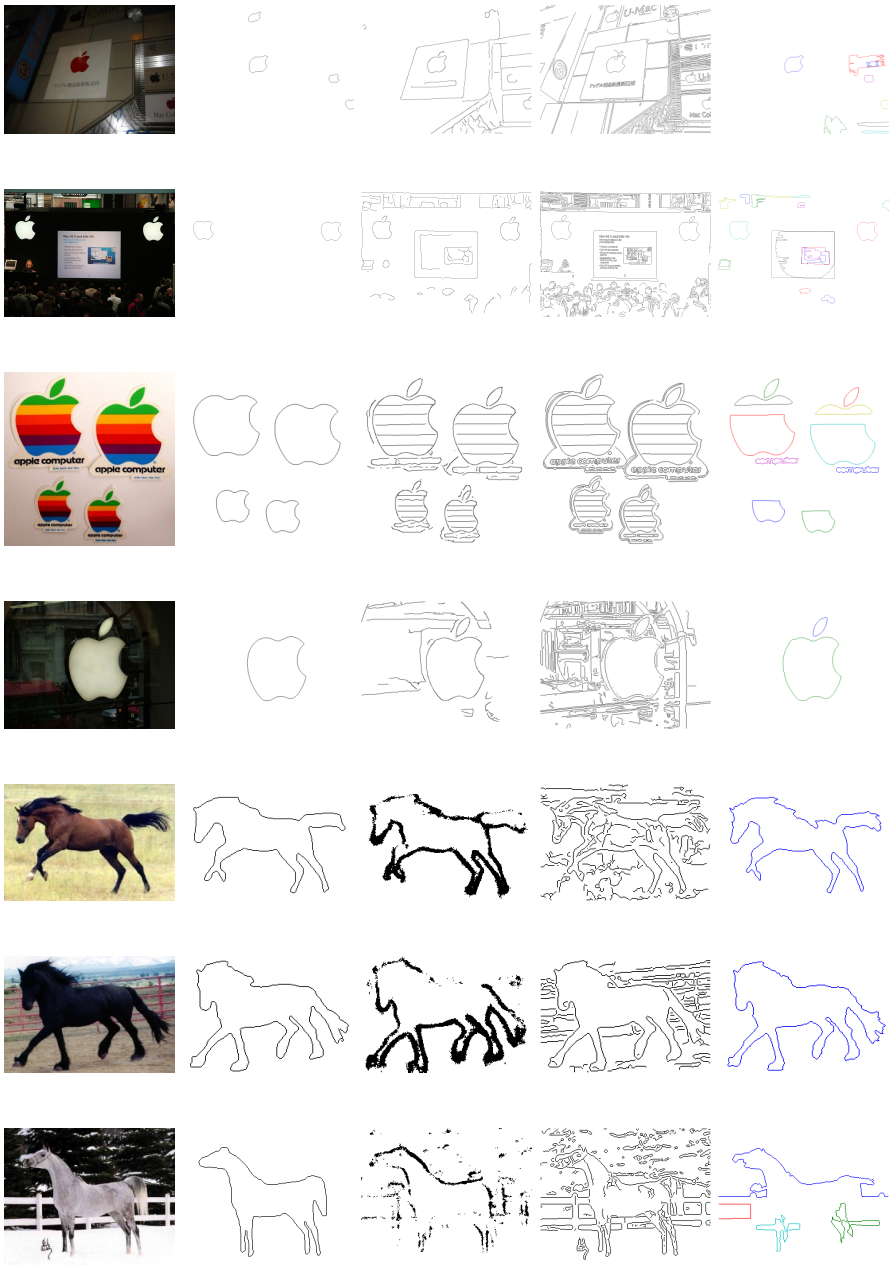


Figure 2.4: Results for the ETHZ Shape class *applelogo* [66] and Weizmann horses [29] given by the input images, the ground truth, the Berkeley [137] and Canny edge [35] responses, and our boundary detection labels. Best viewed under zoom.

The advantages of our algorithm are clear when looking at Figure 2.4, which shows some examples of our stable boundary responses for the ETHZ Shape classes. Our boundary detector produces far less noise, since only stable boundaries are returned,

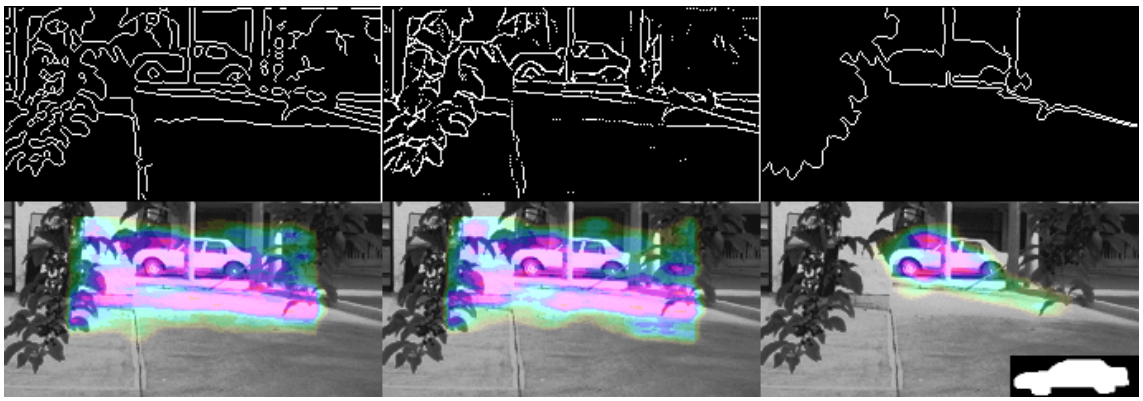


Figure 2.5: First row shows edge detection results from Canny, Berkeley and our boundary detection method. Second row shows corresponding vote maps when oriented chamfer matching is applied on the responses using a car shape prototype.

which have a strong support from underlying regions. This removes a lot of cluttered results, which are present in other edge responses only considering local pixelwise evidence. Occasionally some contours are missing, which for example are present in the highly dense responses of Canny. This is reflected in the lower recall of our method, nevertheless our precision is consistently higher. In general, the precision values are very low since our evaluation strategy is very strict. Only boundaries for the respective object category are marked as ground truth, and thus any other pixel response is declared as a false positive. This is less severe in the case of the Weizmann horses, since the horses are the only dominant object in the images without much background. In comparison, the images of the ETHZ dataset contain much more background information including strong discontinuities in depth, texture and illumination.

A further advantage is the implicit labeling returned by our approach. In contrast to the edge responses from Canny or Berkeley, our boundaries are already connected in a sequence and uniquely labelled. This provides a great benefit, since no perceptual grouping post-processing is required to dissect contour from the clutter, group their continuity across T-junctions, and finally link them into ordered lists. Linking contour responses at T-junctions leave three or more contour parts of the original contour behind. The holistic underlying shape is lost unless even more costly perceptual grouping is performed. Our boundary detector directly provides these grouped results, as we implicitly consider regions as perceptual units for the boundary extraction.

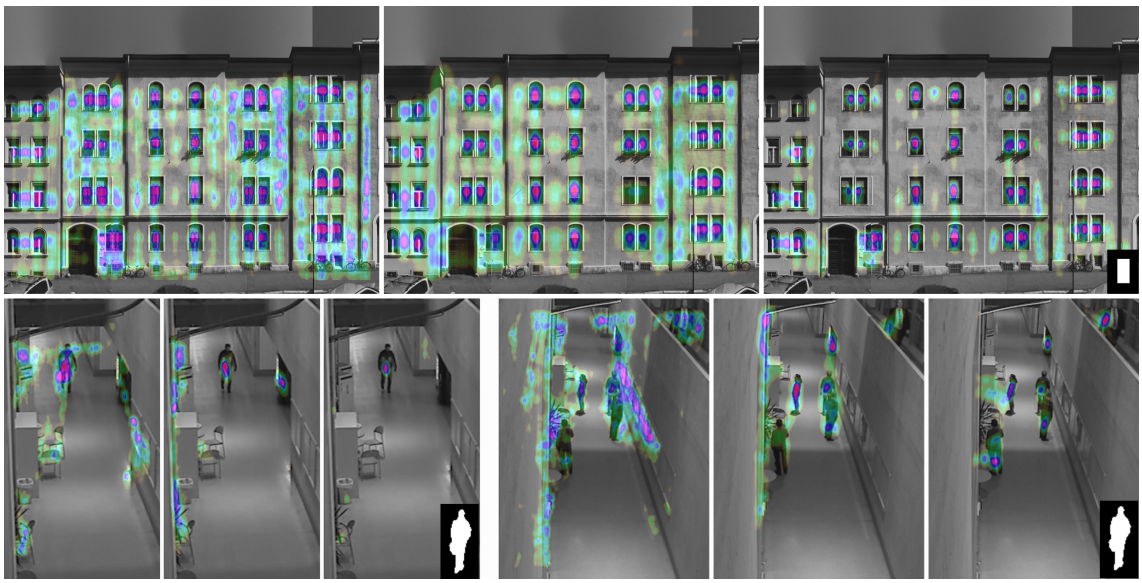


Figure 2.6: Vote maps returned by oriented chamfer matching using a window (first row) and human (second row) shape prototype. First result is always from Canny, second from Berkeley and third from our boundary detector.

2.4.2 Oriented Chamfer Matching

As a second experiment, we evaluate a shape-based object detection approach on our novel stable boundaries to demonstrate the increased performance in precision, recall and speed. Since many of the state-of-the-art recognition approaches [80, 182, 183] implicitly rely on local shape template matching by chamfer matching to localize objects in images, we adopt this method for the experiment. Chamfer matching strongly relies on well-connected and undisturbed contours, where clutter has a strong adverse effect on the recognition performance. Therefore, improving chamfer matching performance is a key research area, where our novel boundary detector provides new insight due to the reduction of clutter through region stability.

We show the qualitative detection performance on several image types for detecting categories like humans, cars or windows. The corresponding shape prototypes are manually drawn or chosen as mean shape when ground truth was available. As matching method we use oriented chamfer matching in [182]. For comparison, we additionally evaluated on Canny and Berkeley edge responses. Figure 2.5 illustrates detected contours and voting maps returned by oriented chamfer matching for the three methods using a car prototype, and Figure 2.6 shows results for localizing humans and windows.

As can be seen our detector returns far less noise while retaining the most important contours for localization, which is illustrated by the much more precise vote maps.

Further, we evaluate the quantitative detection performance on the ETHZ Shape classes dataset [66]. It consists of five object classes (applelogos, bottles, giraffes, mugs and swans) and a total of 255 images. All classes contain intra-class variations (especially giraffes and mugs) and significant scale changes. The images sometimes contain multiple instances of a category and have a large amount of background clutter. For evaluation we follow the protocol defined in [66, 129], where the hand-drawn prototype is searched in the images of the category and evaluate bounding box overlap by the 50%-PASCAL criterion. We use the fast directional chamfer matching framework [129] and only exchange the type of contour detector used for the distance transform. The parameters for the framework are numDir = 70, dirCost = 0.8, maxEdgeCost = 60, and the detection threshold = 1.0 for limiting number of detection hypotheses returned.

Figure 2.7 shows the precision / recall plots for the five categories using the 50%-PASCAL criterion. Although chamfer matching for fixed template models has severe limits on a challenging dataset like the ETHZ, the performance curves show the benefit of our novel boundary responses. In all categories our stable boundaries reduce the clutter and thus provide better contours for detection than a standard Canny response. This results in significant increase in performance and speed.

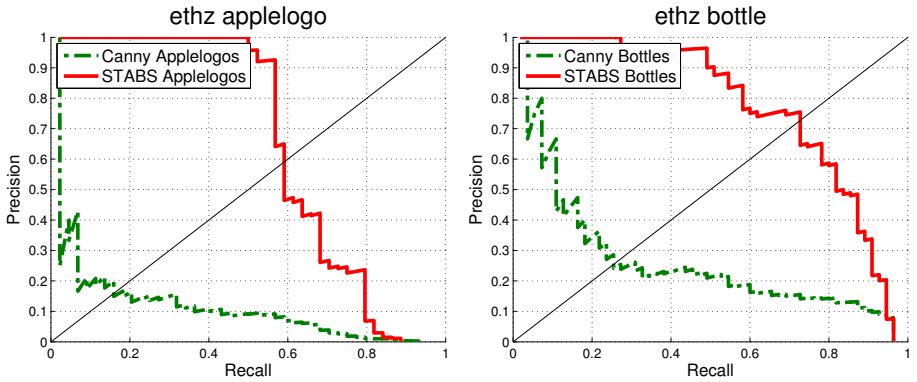
2.5 Conclusion

In this chapter we introduce a novel boundary detection algorithm for the purpose of shape-based object localization. The benefit of our detector is the integration of mid-level context information in terms of support by connected regions. It is based on the analysis of a hierarchical data structure denoted as component tree containing connected regions, which are separated by high gradient values at different magnitudes. We detect the most stable nodes within the component tree, which are returned as labelled boundaries preventing cumbersome post-processing and perceptual grouping required by other methods. We show that our method matches the performance of a supervised learning method and the speed of a standard low-level method.

Experimental evaluations using an object contour benchmark on two well-known object detection datasets, namely the ETHZ Shape classes and the Weizmann Horses, demonstrate the applicability and performance benefits of our method. Our method accurately removes noise and clutter and delivers cleaner, more precise and still rich

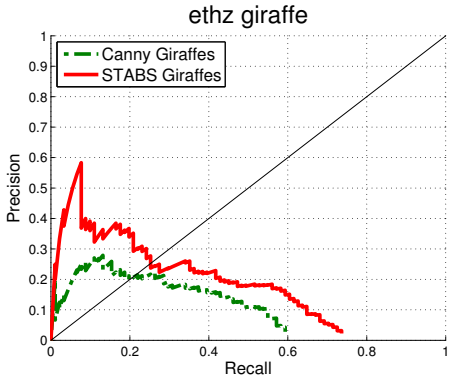
contour responses. The implicit connection to underlying regions eliminates the need for common perceptual grouping, as the regions are used as perceptual units and hence its boundaries are available as linked sequences.

Future work will focus on evaluating our detector in different object localization frameworks and on investigating the benefit of integrating color and texture information. Further ideas are to specifically learn the appearance for certain classes as new input to improve the contour detection results. We are also investigating the integration our mid-level region information into other contour extraction methods.

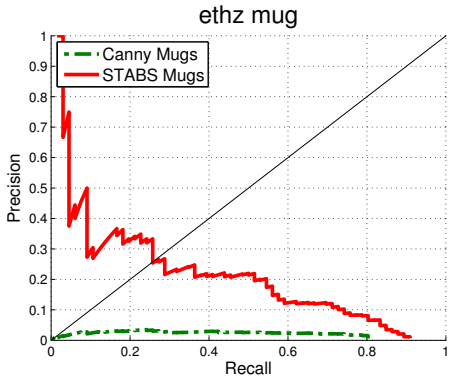


(a)

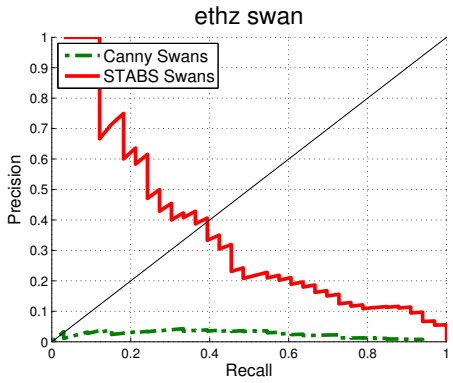
(b)



(c)



(d)



(e)

Figure 2.7: Detection performance for chamfer matching on the ETHZ Shape classes as precision/recall curves using the 50%-PASCAL criterion. Our novel stable boundaries of region significantly reduce clutter and thus outperform the standard Canny approach.

Connectedness is a fundamental powerful driving force under-exploited in object detection.

Vittorio Ferrari

3

Structure in Partial Description

In this chapter we propose a novel structured description of shape. The term shape is used to describe the perceived holistic geometric layout of points of an object. Related work in this field has a long history. We provide an overview of representations, characteristics and notations of shape in Section 3.2 and review related work for shape description in Section 3.3. Our description is inspired by the psychological principles of good continuation and collinearity. We thus focus on the continuity of points and our representation of shape is defined as an ordered sequence of points. This structure in continuity covers a wide range of shape representation, which includes local fragments, salient contour parts and goes up to closed region outlines of an object. Since each point has a unique predecessor and successor, it allows us to take measurements in a structured fashion. The novel description is called *Structural Measurement Descriptor* (SMD) and is discussed in Section 3.4 with regard to transformation invariance, hierarchy, partial description and robustness to noise, articulation and occlusion. In the following we derive specific instances of the general *Structural Measurement Descriptor* for closed regions given by binary masks, or arbitrary long contours and local fragments of shape in cluttered images, as shown in Section 3.5. Experimental evaluation demonstrates better classification performance of our description in Section 3.6.

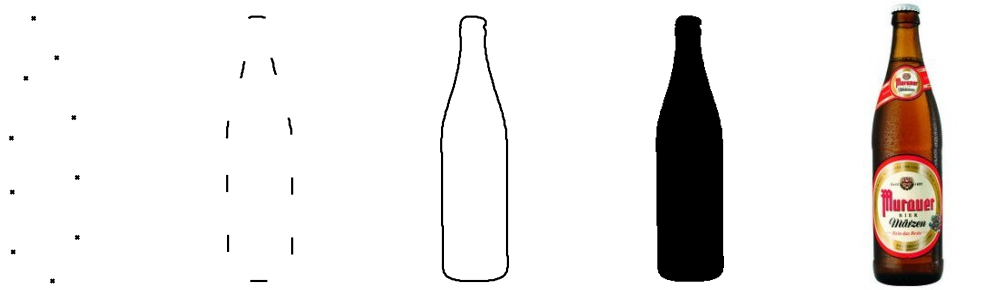
Contents

3.1 Introduction

Visual perception consists of many cues which make up the perception ranging from color, texture, shape and so on. Each of these cues carries information, which depending on the specific object or category, will vary in its importance. For example, when looking at the desk in front of you and the objects lying on top of it, what distinguishes object categories and what identifies specific object instances? It is the color, the texture or the shape of these objects? For a large number of objects, it is the shape which will be the most important and common cue to identify the object category. In this chapter we focus on the description of the shape of either regions, contours or fragments. We introduce a class of descriptors denoted as *Structural Measurements Descriptors* (SMD), which are used to encode the shape of the underlying ordered sequence boundary points. In the following we will describe seven characteristics of shape descriptors and discuss their benefits. The SMDs build a general description, which is more powerful than previous shape descriptors in terms of discrimination, hierarchical decomposition, and partiality to handle occlusion and articulation. These advantages stem from exploiting the underlying structure of the points given by their continuity and connectedness.

3.2 Representation and Notation

The most important part of designing a shape description is the choice of the shape representation. It has a significant effect on how the description and later matching are performed. There are numerous ways how shape can be represented. Figure 3.1 gives an overview of the most popular forms of object representations. The representations



(a) Points (b) Fragments (c) Region (d) Silhouette (e) Texture

Figure 3.1: A range of 2D visual perception in descending levels of abstraction: a) sparse set of unordered points, b) local fragments with limited spatial extent, c) contours as continuous connected parts, d) the boundary of a region, e) the binary silhouettes encoding figure/ground assignment, and f) the full richness including shape, color and texture of an object.

range from unordered sets of points over fully enclosing region mask to the full richness when color and texture are available.

Starting from the least detailed representation, an unordered set of points is exactly this - a set of points without explicit order or structure adhered to. It is difficult to extract a meaning as this point set is only hinting at the original shape. The human perception groups the points trying to make sense out of the unordered chaos, for example by grouping and connecting. This follows the Gestalt principle of continuation [104], which states that the human eye is compelled to follow lines and curves. This flow along a line enables us to complete missing information. It also helps us to connect the dots. Ferrari stated that *"connectedness is a fundamental powerful driving force under-exploited in object detection"* [66]. The next representation hence are small connected fragments, which hint at the shape by including connectedness of a few ordered connected points. Biederman [22] describes in his theory of recognition by components the principles of non-accidental relations introduced by Lowe [130], such as collinearity and curvilinearity of points. These state that this connectedness is a non-accidental feature, which is derived from the view of higher dimensional object. Further up the continuum are local contours, which capture larger parts of the shape by increasing the local size and connectivity to bring more structure into the representation. When the contours are closed, they form the boundary of a region. It is a single closed representation cap-

turing the holism of the shape. As a binary silhouette separating figure and ground, shape is encoded as a filled region, which includes the boundary as well as its area. The final level adds color and texture, which then provide the full richness of an object in 2D. In general, one is looking for a description of shape, which captures the following invariances and characteristics.

- **Rigid transformation invariance** is a concept in Kendall's theory of shape [102], which defines the deformation a shape must be able to undergo and still be perceived as the same shape to be invariant to rigid transformations. These rigid transformations are composed of translation, rotation and scale, which still keeps the perception of a shape. Hence, any description should be able to provide invariance to similarity transformations, unless otherwise required.
- **Non-rigid transformation insensitivity** requires that the description is insensitive to small deformations caused by transformations, which slightly change the perception of shape but still keep its essence, for example, stretching or shearing.
- **Sequence** is a main driving force in human perception, which exploits the continuation and proximity of a set of points to group them and hence derive an ordering for point sequence.
- **Hierarchical description** implicitly includes local and global information within a single description. A descriptor is thus the combination of multiple levels of scale. It also determines that the description has a common representation for an object or its parts, for example as in regions, contours, and fragments.
- **Partial description** allows the self-containing description of parts of the shape within a single description. A descriptor is thus the division into its individual parts, the sum of these parts and more so its configuration. This benefit enables holistic grouping of parts and handles occlusion and articulation.
- **Articulation and occlusion** are deformations, which affect the shape due to changing configuration of parts or noise disturbing parts of the shape. A description should be able to handle the presence of articulation and occluded parts.
- **Detachment of clutter** is the property that the context used to describe the shape is solely based on the shape itself. Proximate other information in neighborhood reduces the focus on the shape and often belongs to a different part, background clutter or noise.

In this chapter we focus on the continuity of contours and our representation of shape is defined as an ordered sequence of points. This structure in continuity allows a wide range over the continuum of how shape may be represented, which may range from local fragments over contour parts to fully closed outlines of a shape. Hence, our representation consists of a sequence of points, which is derived from the boundary of a region or parts of it. They are sampled points without a specific sampling strategy and are denoted as boundary points. Each of the points has a unique predecessor and successor, which allows us to take measurements in a structured fashion. Throughout this chapter, the word shape is used as a general term to describe the perceived holistic geometric layout of points of an object. It will be used when the general meaning of the internal and external form of an object is expressed. For specific representation and description, we use the following terminology and notation.

Point is any single point somewhere on a shape and denoted as $\mathcal{P} = \{P_x, P_y\}$, where P_x and P_y are coordinates of the point.

Landmark point is a point, which distinguishes itself from the other points by a special characteristic, such as low or high curvature, and is denoted as $\mathcal{L} = \{L_x, L_y\}$.

Boundary is an ordered sequence of points and is denoted as $\mathcal{B} = (P_1, P_2, \ldots, P_N)$, where P_i is a single or a landmark point and N is the number of the boundary points in the sequence. This boundary may stem from a closed silhouette or a partial image edge.

Region is a special case of a boundary sequence denoted as $\mathcal{R} = (P_1, P_2, \ldots, P_M)$, where the boundary is closed, i. e. P_1 and P_M are the identical points.

Contour is the standard case of a boundary sequence denoted as $\mathcal{C} = (P_1, P_2, \ldots, P_M)$, where the boundary is open, i. e. no points are used twice from the boundary points.

Fragment is a local form of the boundary denoted as $\mathcal{F} = (P_1, P_2, \ldots, P_K)$, where P_1 to P_K is again a connected subset of the boundary points. Thus it is also an ordered sequence, however of known finite length. It contains only a fixed number of points K, which is significantly less than for a contour ($K \ll N$).

Silhouette is a binary representation of the region of a shape denoted as $\mathcal{S} = \{P_1, P_2, \ldots, P_A\}$, where each point in the set is defined as a foreground point. It expresses the area of a shape without explicitly specifying the sequence or individual points on the boundary.

Throughout this work, shape is hence defined as a set of boundaries $\mathcal{B}$, which itself are composed of a sequence of points $\mathcal{P}$. In the related work the terminology and representation vary strongly, and its analysis in the next section will be grouped into the notions of global, semi-local and local capturing of the shape instead. This will hopefully give a good overview of the numerous approaches.

3.3 Related Work

In this section we summarize the related work for the most popular types of shape description. We will focus on the range of descriptions and provide an analysis according to the seven characteristics, which one is looking for when describing shapes. As there is a range of ways to represent shape, ranging from the silhouettes of a region, over continuous curve representations to sets of unordered points, there is an equal multitude of ways for description of these shape representations. Hence, we divide the related literature into three categories: global, semi-local and local approaches.

Global methods analyze the overall shapes of the input objects by defining a global matching distance. This has a holistic notion of shape, however suffers when parts of the shape are missing or articulated. Semi-local shape approaches are the pendent of local interest points, which describe the underlying information within a spatially limited support window. This lifts some of the constraints on occlusion, articulation, etc. However, it requires shape category models to fill the missing global structural information. Local approach go one step further and extract shape information from a single point. This renders the description further invariant towards effects of occlusion and articulation, yet requires strong higher-order graphical models to bring global structure back into the shape.

3.3.1 Statistical global representations

Global statistical representations for shape have a long history. They capture the global spatial layout of the object and maintain a holistic representation. However, these often

require the object of interested to be well-segmented and hence are susceptible to noise, occlusion and articulation when dealing with real life images.

Examples are coordinate frames [26], curves [178], image moments [96, 103], Fourier descriptors [124], skeletonic graph-based representations like Medial Axis [25], Shock Graph [186], Symmetry Sets [111], Poisson Shape [85] and more complex methods utilizing optimization for adapting to the image such as the Active Shape Models [40]. In the following we will outline some of the basic underlying concepts for global statistical shape representations.

3.3.1.1 Bookstein coordinates

Bookstein [26, 27] derived a method to statistically describe shape, once they are transformed to a common coordinate frame. Bookstein introduced the baseline concept, where a local coordinate frame is constructed on two landmark points. The first landmark A is set as the origin (0,0) and the second landmark B is rotated and scaled so that its coordinates lies at (1,0). This creates the basis for a rigid transformation. The remaining boundary points are then transformed in the same way, which provides a representation that is invariance to scale, rotation and translation (similarity transforms) as defined by

$$p = u \times p_x^s + v \times p_y^s \tag{3.1}$$

where the scalar quantities u and v define the similarity transformation given by the baseline landmark points. This single coordinates frame transforms all remaining boundary points into a common frame, which can be used for statistical measurements. The choice of the baseline however may strongly effect the global representation, and hence the selection of the initial two landmark points is critical.

3.3.1.2 Procrustes Analysis

Procrustes analysis [84] is a method for statistical analysis of shape by comparing the points after they have been aligned by translation, scale and rotation. The two sets of point can be independently normalized by translation and scale by shifting to the mean point and normalizing by the mean distance, respectively. Rotation alignment requires a reference orientation, for which usually one of the shapes itself is used. The two sets of points are overlayed and a difference between the superimposed points, for example

Euclidean distance between the coordinates, determines how well two shapes align. For rotation alignment, the optimal orientation is given by the smallest difference between a rotated shape and the reference shape. This distance is then used to describe the quality of matching between the two shapes by

$$\sum_{i=1}^{N} = ||\mu - \beta_i \times exp(i \times \theta_i)(z_i - \gamma_i \times 1_k)||^2 \tag{3.2}$$

where β_i determines the relative scale, θ_i the relative rotation, γ_i the relative location, and μ which represents a centered and scaled set of points. This approach only uses global statistical information to align two sets of points, however ignores any pairwise information. Hence, the corresponding alignment is stretched and scaled to best fit the reference point set.

3.3.1.3 Image Moments

Image moments are a method that captures the statistical properties of a shape by viewing it as a probability density function over its range. The range is an image of a shape, in which each pixel is equally important (in case of a binary image) or weighted (according to gradient intensities). The idea is to calculate moments, which capture geometric properties of the shape by characteristic values such as area, center, centroid, covariance, and so on. These values are then combined into higher-order moments, which express invariance towards translation, rotation, scale, etc. Well-known sets of such moments are described in the centralized moments, the Hu invariant set [96], Zernike moments [103], and many more.

This approach provides a compact description of a set of points under similarity transform (Zernike), however also requires the full shape to be segmented. The descriptions are thus not robust to occlusions or articulation. If a part is missing, the description is changed, for example by shifting the center of mass and decreasing the total area.

3.3.1.4 Fourier Transforms

Fourier Transforms [188] of shape have been used to describe closed boundaries around a region. The two-dimensional coordinates are used within a complex plane. Following the sequence of points along the boundary (in an anti-clockwise direction), a complex function z(t) is obtained as the sum over coefficients T_n called the *Fourier descriptors* for this boundary. They are influenced by the boundary shape and its initial starting point.

Choosing an appropriate coordinate system renders the *Fourier descriptors* invariant to translation and rotation. This has been employed for digital character recognition [185].

Since small changes the boundary will produce global changes in the *Fourier descriptors*. Wavelets are a local variant in both frequency and time, and therefore analyze the shape of boundaries at different scales. Their representation uses a smaller basis function, for example, the Haar wavelet function [91], which is more robust to changes.

3.3.1.5 Axis-based representations

Graph-based representations have been introduced as the medial axis by [25], which are a skeleton representation of the binary shape. The medial axis is a set of points at equidistance from the outer boundary of the shape. This axis is determined by searching for the centers of maximally sized circles within the region of the binary shape. Straight skeletons [2] are an alternative formulation to the curved segments of the axis by deriving straight lines. These skeletons are formed by moving the point on the boundary of the shape into the region at constant velocity. Both methods require well-segmented object shapes and the final description is very sensitive to occlusion and noise.

Shock graphs [186] view the evolution of such axes during their formation, and describe the singularities in a graph representation, for example the corners, bridges, lines and points. The graph is representation as a tree and the edit events (adding or removing nodes) are defined as the similarity when matching two shapes [177]. Matching is be viewed as finding subtrees of the graph and thus occlusion handling is now possible. However all the methods require a closed region as input to infer the axis and are not applicable to contours.

3.3.2 Local feature representations

Local representations of shape also share a long history of research. Their main benefit lies in the independence of the features by describing only a small local area around a boundary point. This enables the features to be independent of global properties or other information in the local neighborhood of the shape. While it can be difficult to robustly determine global properties such as center of gravity or global orientation, local approaches are robust to occlusion and articulation of the shape. In the following we outline some of the local features based on turning angles, tangents, normals, distances and arc lengths, which are the basic building blocks for local shape analysis. However,

their strength is derived from a combination of multiple local features and graphical models constraining the global geometry [20, 63, 122, 195].

3.3.2.1 Curvature and Tangents

The most frequently used local features are the curvature (also referred to as turning angle) and the tangent space (referred to as the slope). Since these features are computed locally, they determine a single value for a single given boundary point. For an entire shape, the sequence of local features form a 1D signature. Such a signature is a 1D description of following the region boundary of a shape. This results in a compact and local description of the shape.

The information extracted by turning angles is the change in direction encoded in radians. This bears similarity to the well-known chain codes, which quantize the change in direction into eight angular directions [76]. An alternative description defines the rate of change of direction, which requires a support window to calculate the rate of change. Essentially this determines a derivative of the curvature. Curvature scale space (CSS) analysis [144] uses successive Gaussian blurring to determine inflection points by zero-crossings along the blurred outline. These extrema are used for detecting landmarks points at high and low curvature. On the other hand, tangents represent the normal vector at a given boundary point. The tangent is calculated by the slope of the shape at the boundary point.

The notion of curvature is intuitive and in theory robust to deformation induced by articulation or occlusion. However recent work by Arkin et al. [11] and Basri et al. [17] stated that the invariance due to local representation is too high. Hence, for example, a single change in curvature can drastically alter the outline of the shape. Turney et al. [199] thus used the tangent by slope and additionally arc length as local representation for shape. Similarly, Schindler and Suter [172] proposed a probabilistically motivated shape distance based on tangents which requires a closed contour in the images. Chen et al. [37] defined the curvature by the turning angle (TA) calculated between a three consecutive points (P_{i-1}, P_i, P_{i+1}) on a boundary. The normal at middle point P_i dissects the outline of the other side of the shape at point P_i'. The distance $\overline{P_i P_i'}$ is called the Distance Across the Shape (DAS). The distance description is normalized by its maximal distance, and the TA and DAS features are then linearly combined.

The set of points used in this description are boundary points on the outline of the shape. The landmarks are selected at high/low curvature points and at maximal

distance between each landmark point. This reduces the number of points and hence the description, however it requires a stable extraction of landmark points to make the description repeatable. Further, this local approach lacks information about the global shape, and is thus more sensitive to noise and articulation.

3.3.3 Semi-local representations

Semi-local representations capture local features, however over a larger support window marginalizing over the denser observation. The semi-local support allows moving away from the individual well-segmented object representation and allows part description of objects. As a consequence the descriptions are less susceptible to cluttered edges by spatial distance weighting or independently representing each detached contour edge. The main two paradigms are the marginalization of the multiple local descriptions into distributions, and the use of the complete set as redundant observation combined with a graphical model. In the following we will outline some semi-local methods and how they are used for partial shape description.

3.3.3.1 Geometric Hashing

Geometric Hashing is a method proposed by Wolfson and Rigoutsos [208] for representing two-dimensional shapes by encoding each pair of points by an affine-invariant basis. This basis is another pair of points, which rotates the local coordinate frame and allows invariance towards similarity transformations (scale, translation, rotation), similar to Bookstein coordinates as

$$p - p_0^s = u \times p_x^s + v \times p_y^s \tag{3.3}$$

where $p_0^s = (p_i + p_j)/2$ is the center between the point pair (P_i, P_j) defining the baseline. Given a base pair, the scalar quantities v and u allow a description invariant under similarity transformation. This representation is repeated for all combinations of base lines and remaining points. The quantities v and u are further used as a 2D hash index to quantize and store the set of points. In the matching stage, boundary points are detected and hashed using randomly selected base pairs. Each matching hash entry from the database then votes for the matching baseline until an object can be localized. This method has been applied in for 2D and 3D alignment of shapes [140] as well as

in other fields, for example, protein alignment in bio-informatics and star constellation search in astronomy.

3.3.3.2 Pairwise features

Pairwise interactions of simple features are proposed by Leordeanu et al. [122] to model how two contours preserve their geometric relationships. The method uses seven measurements such as orientation angles, distances and normal orientation to encode a feature vector between two landmark points. Each landmark point resides on a contour associated with a normal vector. For a pair of landmarks (P_i, P_j) they define a translation-invariant feature by

$$e_{ij} = \{\theta_i, \theta_j, \sigma_{ij}, \sigma_{ji}, \alpha_{ij}, \beta_{ij}, d_{ij}\} \tag{3.4}$$

where d_{ij} is their distance and β_{ij} is the angle between their normals θ_i and θ_i. The remaining angles α and σ encode orientation information about the line between the two landmark points.

The individual landmark points sampled on the contours are used as model parts in their learning and detection stages. The power behind this approach comes from the grouping of multiple simple features to encode their pairwise geometry in a graphical model. Each pair (P_i, P_j) describes a local feature which encodes part of the object shape independently. Hence, they are operating with a set of local landmark pairs, which in turn then defines a category model and enables it handle occlusion and articulation.

3.3.3.3 Shape Context

Shape Context is a method proposed by Belongie et al. [18, 19], which captures the effect of the neighborhood for a specific location of the shape. The idea is that the context encodes the configuration of the boundary points relative to a reference point. Instead of calculating pairwise properties of the points, the shape is stored in log-polar space where the context of a given boundary point. The space around the boundary point is divided into d angular directions and r logarithmic spaced radii. Each point is then assigned to one of these $r \times d$ bins and the description is a histogram by

$$h_i(k) = \#\{P \neq P_i : (P - P_i) \in \text{bin}(k)\} \tag{3.5}$$

where P_i is the given reference boundary and P are the remaining points. The log-polar space allows a linear increase in position uncertainty, which makes the descriptor more less sensitive to faraway boundary points. The histogram counts the frequency of a certain shape context leading translation invariance. Scale invariance is either achieved by normalizing all distances by the mean distance of all pairs of points, or by a support window to reduce the set of points. Rotation invariance is not typically used, as it requires the selected of a dominant direction, which sorts the directional bins and thus normalized all measurements by a common basis.

Since the Euclidean distances are used to select the log-spaced bins, the description is sensitive to articulations. An adaptation of the method known as Inner-Distance Shape Context introduced by Ling and Jacobs [126] uses the inner distance between two points moving only inside the shape. This inner distance brings the benefit of invariance to articulation as the description is now a canonical form of the shape.

Each boundary point in the shape is described by a separate descriptor. Since the complexity of the descriptor is constrained by the histogram binning, the set of boundary points used densely without resampling. Belongie et al. mention that using only inflection and extrema points will populate the histograms less densely leading to a less distinctive robust description.

A further extension is that of Berg et al. [20], who propose a continuous form of the *Shape Context* denoted as *Geometric Blur*. Instead of using binary edge points, the image gradients are accumulated in the histogram and the spatial uncertainty is realized by successive Gaussian blurring.

3.3.3.4 k Adjacent Segments

The k-Adjacent Segment (kAS) is a method by Ferrari et al. [63, 66], which connects neighboring line segments to a local description. This is done by building a contour segmentation network (CSN) to define a graph of all edges in an image, abstracted by straight lines and linked at junctions. A single segment is one straight line, which is defined by its endpoints, centroid, angle and length. The k-Adjacent Segments (kAS) is a higher-order composition of k such segments leading to mid-level complexity of local shapes, for example as L- or T-shapes (for k=2) or Z- and U-shapes (for k=3). Ferrari et al. defined a scale, translation and rotation invariant description for kAS and used these for object detection by

$$kAS(k) = \{\frac{r_2^x}{N_d}, \frac{r_2^y}{N_d}, \ldots, \frac{r_k^x}{N_d}, \frac{r_k^y}{N_d}, \theta_1, \ldots, \theta_k, \frac{l_1}{N_d}, \ldots, \frac{l_k}{N_d}, \} \tag{3.6}$$

where $r_i = (r_i^x, r_i^y)$ is the vector between midpoints of two segments, the distance N_d is the maximum of all midpoint distances, and θ_i and l_i the orientation and length of the segment. Due to the normalization factor N_d, the descriptor is thus scale and translation invariant.

One benefit of this description based on segments is the detached encoding of part of an object shape without being affected by nearby clutter or noise, unlike local context-based methods such [18, 179].

However, the main benefit lies in the connectedness of the local segments forming more complex shapes. Instead of an independent local feature, multiple segments form more complex local shapes. This way the kAS provide abstract landmark points by their mean centroid, which define then further are grouped to defined a shape model. However, the length of any segment is constrained by its local curvature, which limits the spatial extent of a segment. Since each segment represents a straight line abstracting the actual underlying shape.

3.3.3.5 Shape Tree

Shape Tree is a method by Felzenszwalb and Schwartz [61], which builds a hierarchical description of a contour inspired by Mokhtarian and Mackworth's curvature scale space [144]. The hierarchy is built by spitting a contour at its mid-point and creating a binary tree. The left and right nodes are filled with the corresponding divided contours. Hence each contour is placed at a new level starting with the midpoint of the previous level. This representation has the benefit of encoding global (nodes further up the tree) and local (lower nodes) information about a shape. Each node stores the relative coordinates of the midpoint and its two endpoints. Further, Bookstein coordinates on these relative coordinates are used to achieve invariance towards similarity transformations. Hence, the representation is very sparse for global nodes and more detailed for lower nodes describing only very local parts of the shape. Felzenszwalb and Schwartz used Shape Trees for matching objects by measuring the deformation between two Shape Trees by recursive calculation the matching cost each level.

3.3.3.6 Chord distributions

Chord distributions are introduced by Cootes et al. [40] as a method to obtain trainable parametric descriptions of shapes. A chord is a line joining two boundary points on a shape. Instead of viewing it as a baseline for a coordinate system, the chord itself is described. The description of a chord is either performed by a distance measurement. Multiple such chord descriptions for a shape are then combined into chord length distributions (CLD), which measure the global shape and frequency of the chords. Further, a shape model is built by describing the covariance between a chord and the chord distributions.

In a similar fashion, the line between two landmark points has been used by Saber et al. [168], who also sort the landmarks and Arica et al. [10], who denote them as beam angle statistics (BAS). A beam angle is the angle of a line between a reference point and all other points on a shape. Multiple beam angles are summarized in a compact histogram representation. Payet and Todorovic [159] presented an approach for object detection by mining repetitive spatial configurations of contours and representing them as sequences of weighted beam angle histograms.

The description is a global combination of multiple semi-local chords. Such distributions combine multiple observations into a robust statistical measurement, however lose all knowledge about the actual location of the chords.

3.3.3.7 Chordiograms

Chordiogram is a method proposed by Toshev et al. [194, 195], which is a joint histogram capturing the distribution over all chords with the shape of an object. Each chord $\overline{PQ}$ is described by four geometric measurements. The information collected is the chord length, its orientation, and normal vectors pointing inwards. These simple features are quantized into bins and stored in a histogram as

$$ch_k = \sum (p,q) | f_{pq} \in bin(k) \text{where} f_{pq} = (r_{pq}, \theta_{pq}, \omega_p, \omega_q) \tag{3.7}$$

which gives a global description of the entire shape by adding each individual chord to the joint description. The benefit of this representation is the explicit representation of figure/ground by means of the interior oriented normals. This is used in the matching stage to guide perceptual grouping and increase the robustness over pure contour description.

3.3.4 Summary

The range and characteristics of the related work is summarized in Table 3.1. The overview shows that many approaches fulfill Kendall's theory of shape transformation invariance in terms of invariance to similarity transformation. However, most approach aim for one level of scale in terms of local or global description and do not construct hierarchies. Further, notions of self-containing description which enables the partial description leading to occlusion and articulation invariance as well as detachment are rare. In the next section we will describe the family of shape descriptors denoted as *Structural Measurement Descriptors*, which satisfies all of these requirements.

	global	semi-local	local	rotation	translation	scale	hierarchy	detachment	dense	partiality	articulation	occlusion	pairwise
Bookstein	+			+	+	+		+	+				
Procrustes	+			+	+	+			+				
Moments	+			+	+	+			+				
Fourier	+			+	+	+							
Medial Axis	+			+	+	+							
Shock Graph	+			+	+	+	+			+		+	
Curvature			+		+	+		+		+	+	+	
Tangent			+		+	+		+		+	+	+	
Geometric Hashing		+		+	+	+			+			+	
Pairwise Features		+			+	+		+			+	+	+
Shape Context		+			+	+			+				
Inner-Distance SC		+			+	+			+		+		
Adjacent Segments		+		+	+	+	+	+		+		+	+
Shape Tree	+	+	+	+	+	+	+	+		+			
Chord Distribution	+			+	+	+		+	+				
Chordiogram	+	+			+	+		+		+			+
SMD	+	+	+	+	+	+	+	+	+	+	+	+	+

Table 3.1: Overview of related shape descriptor and their properties in contrast to the proposed *Structural Measurement Descriptors* (SMD).

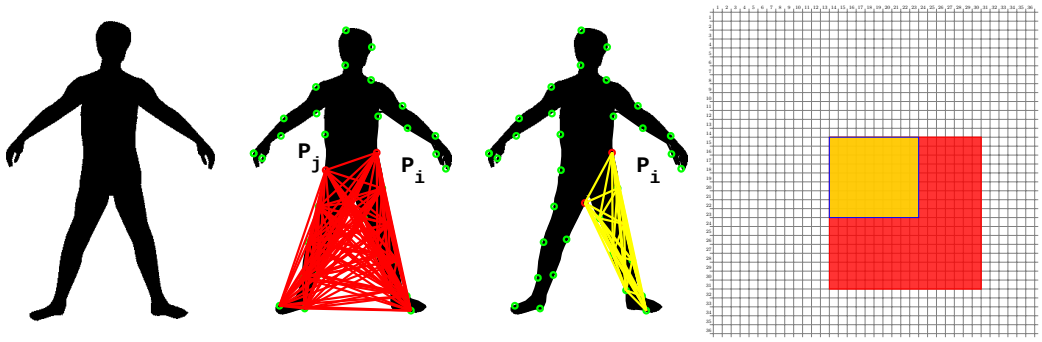
3.4 Structural Measurement Descriptors

In this section we describe the concept for shape description denoted as *Structural Measurement Descriptor* (SMD). The measurements are inspired by simple features such as angles and distances over chords, however combining pairs of chords to form a measurement. Further, we build a new hierarchical representation that allows partial matching along the actual sequence of the contours. The description encodes the shape of a contour by recursively defining its parts. Yet this hierarchy is represented in a flat matrix form, which allows efficient description and later matching techniques. The partial matching is directly connected to this structural measurement descriptions and is not built on matching individual boundary points.

It is our goal to exploit the ordering of the measurements to derive a description and consequential matching which is order-preserving. Hence, the descriptor should exploit the available point ordering information. In comparison, other descriptors, for example the Shape Context or Chord Distributions, lose all the ordering information due to the histogram quantization. Hence they do not consider the influence of the sequence information in the local neighborhood on single point matches. Further, we present a general form for representing pairwise information between two chords, which are defined by two points. Further, our description allows to combine pairs of chords to achieve higher abstractions of the representation. In the following we show examples with angles and distances, however the representation is not limited to these and may include additional information about chord pairs, such as gradient or color profiles [192].

3.4.1 Structured Measurements

The new description is inspired by chord distributions introduced by Cootes et al. [40]. They define a chord as the line joining two points on a shape. Our descriptor uses such chords, but instead of building statistical distribution histograms, we use the relative information between specific chords and exploit their order. This has two main benefits. First, this description holds pairwise information which makes it more robust and can abstract information to undergo invariance to similarity transformation. Second, the description is based on the specific ordering of any relative pair information. Hence, we derive a representation which captures this ordering implicitly. The measurements represent information from very local up to global chords. This builds a powerful hierarchical description. In contrast to related approaches, this hierarchical description



(a) Shape (b) Part A (c) Part B (d) Submatrix

Figure 3.2: The Structured Measurement Descriptor (SMD) is based on deriving information from $N \times N$ chords for each tuple of points (P_i, P_j) on the boundary of a shape. The measurements are designed such that each part of the boundary is also a sub-matrix of the full description.

describes a local boundary point, but also the combination of multiple boundary points within the same description.

Given a boundary $\mathcal{B}$, which is defined by the sequence $\mathcal{B} = (P_1, P_2, \ldots, P_N)$, a chord is the line connection between any two of these points. Since we have the ordered sequence of these points, we can also order any information extracted from the chords. We thus defined a chord by an ordered tuple of points (P_i, P_j). For the first point P_i we can derive N different measurements $\psi_{i1}, \ldots, \psi_{iN}$. Additionally, every other point on the boundary is selected as first point of the tuple, and thus an $N \times N$ matrix Ψ defined as

$$\Psi = \begin{pmatrix} \psi_{11} & \cdots & \psi_{1N} \\ \vdots & \ddots & \vdots \\ \psi_{N1} & \cdots & \psi_{NN} \end{pmatrix} \tag{3.8}$$

which is used to redundantly describe the structure and sequence of the entire boundary. The matrix contains dense measurements between any two points of the boundary and thus captures the essence of the structure we want to measure. Along each sides, we index the order of the points along the boundary. The columns are measurements between an index point P_i and any other point P_j on the boundary. Thus for small differences between the indices i and j, these are points close to each other providing local information. For large differences, the points far apart providing global information. Elements on the main diagonal $\psi_{11}, \ldots, \psi_{NN}$ represent the information between a

boundary point and itself. This description implicitly includes local (close to the main diagonal) and global information (further away from the diagonal).

Further, each part of boundary is an independent part of the matrix description. Given a sequence of boundary points $\mathcal{B} = (P_1, P_2, \ldots, P_N)$, its structural measurement descriptor is defined as in Equation 3.8. A part of the sequence defined by a subset of the points $\mathcal{B}' = (P_k, P_{k+1}, \ldots, P_n)$ where k and n are smaller than N, is defined by

$$\Psi' = \begin{pmatrix} \psi_{kk} & \cdots & \psi_{kn} \\ \vdots & \ddots & \vdots \\ \psi_{nk} & \cdots & \psi_{nn} \end{pmatrix} \tag{3.9}$$

which is a sub-matrix of the complete structural measurement descriptor. Figure 3.2 shows this property by outlining the tuples $(P_{k:n}, P_{k:n})$, which define the chords in only a part of the boundary.

3.4.2 Holism

In this chapter we present a shape description which captures the holism of a shape, partly inspired by the principles of the Gestalt school of perception. Holism is the notation that there is an extra value to a collection of parts. This is perhaps best portrayed by the words of Koffka as *"the whole is other from the sum of its parts"* [104] hinting that there is something different about a entire object than merely its parts. Agreeing with this is also the evidence discovered by Palmer [156] that configuration of parts provides more information than which is contained of its individual parts alone.

In the proposed *Structural Measurement Descriptors* we capture such information by including local parts and configuration between those parts. These configurations are the global aspects of the shape, and are collected further away for the diagonal. Contrary to many related descriptors, our description implicitly includes a hierarchy and configuration of parts. The parts of a boundary are contained within the full description and can be addressed separately. We denote this property as self-containment. It allows to describe and match each part independent of its extent and possible occlusions.

The self-containment is a main benefit which enables partial description. Unlike the global shape description methods, our description encodes each part independently as well as within the full description. Chordiograms [195] achieve this by designing their descriptor the non-negative summation of part description. While this makes it powerful to marginalize over multiple parts and compare a joint description, it also introduces

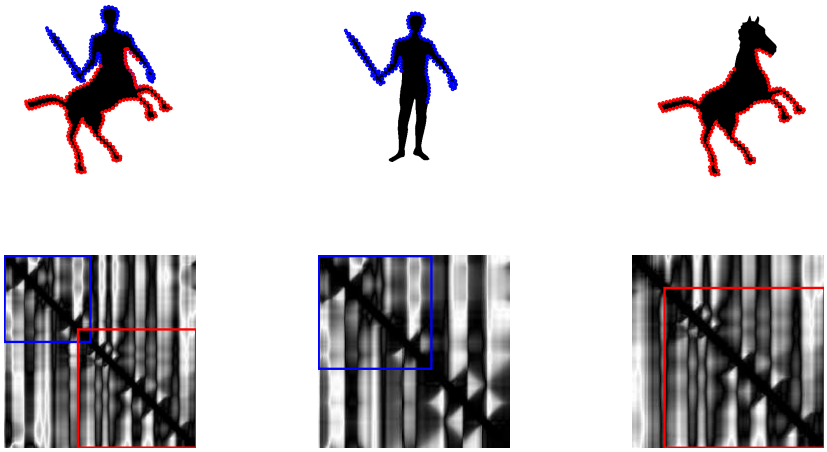


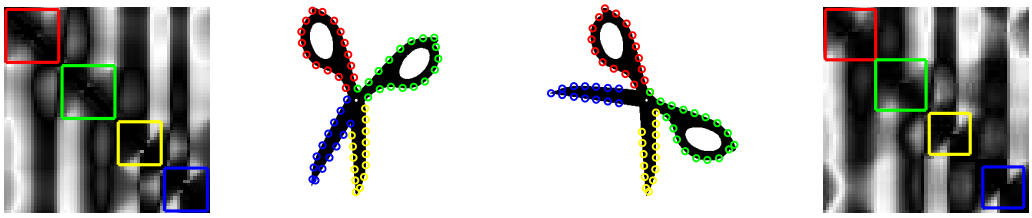
Figure 3.3: The holism of a shape is captured by the Structural Measurement Descriptor. Each part is individually description and captures, however the off-diagonal elements of our description capture the configuration of parts.

limitations of the marginalization. This description does not allow to a reversely extract single parts from the joint description and suffers from noise introduced by articulation of the parts. Out description explicitly encodes each part and extends each part's description as more and more boundary points are connected. This implicitly results in a local and global description, which captures the parts, their configuration and the holism of the shape.

Consider the following example in Figure 3.3 showing three creatures: A horse, a man and a centaur. Partially they are very similar as the torso is used in the horse and the centaur, and the upper body for the centaur and man are both human. Hence, the descriptions of these parts are also very similar and outlined with their respective colors. In the unmarked regions are the parts which encode the configurations. For example, it reflects how the human upper body interacts with the horse torso.

3.4.3 Articulation

This benefit of self-containment also results in a part-based and global description. A common drawback of related work following holism is that the description is affected by articulation or clutter. For example, pure semi-local and global approaches include all information in a descriptor and thus also include nearby clutter. Similarly the whole descriptor is effected when parts of it are changed or removed. Methods like Shape



(a) SMD Closed (b) Closed (c) Open (d) SMD Open

Figure 3.4: Articulation invariance is handled by returning a set of partially matching boundary parts. Corresponding boundary parts are depicted by the same color.

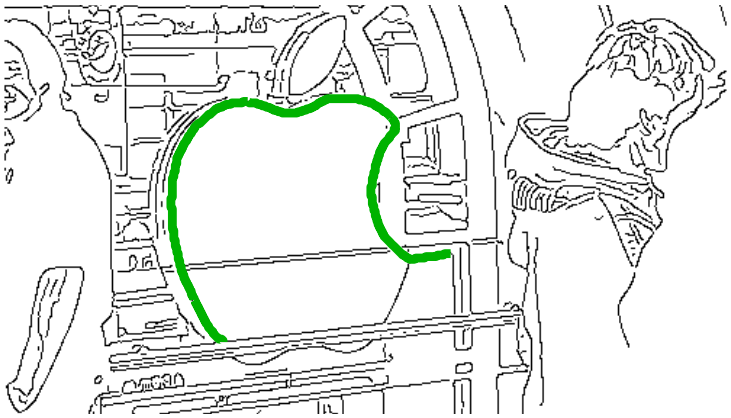


Figure 3.5: Detachment from clutter is an important requirement, as it only encodes part of the shape that is to be described.

Context, k-Adjacent Segments and Chordiogram methods are not able to match articulated shapes because of the non-detachment and articulation-sensitive descriptor. The hierarchy and partial description of our description enables invariance towards these deformations. This ability is illustrated in Figure 3.4 showing two scissors and its four main components of the shape. Due to the self-containment each articulated part is described on its own. However, the off-diagonal elements encode the configuration of the parts. The smaller parts (shafts along the blades or handles) remained the same, hence, their description and configuration remained the same using our approach.

3.4.4 Detachment

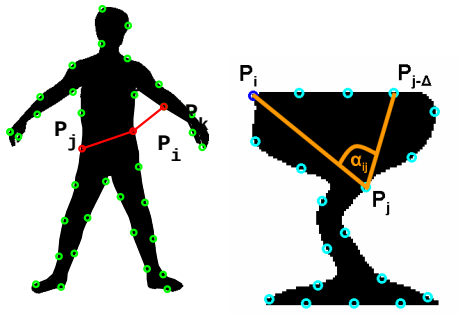
A further benefit is that of contour detachment from cluttered images. The notion of detachment describes how a shape description is influenced by near by shape information, which is not part of the current boundary of interest. A part description is detached, if it is not affected by any other information than its own boundary. Descriptions based on patches or support regions extend the information used for a description to a surrounding area. While this is well-suited for interest points within an object, interest points at the boundary of an object are influenced by background information. Since shape description deals solely with boundary information, any additional information which is not part of the boundary, is obfuscating the description. Figure 3.5 illustrates this detachment by highlighting a single contour of the object of interest (in green). It is separately encoded and therefore is not affected by the near by clutter of background edges in the image.

3.4.5 Features

In the following we describe the sources of information for describing a pair of chords. In this chapter we discuss the two such sources, namely distance and angles, which we use for calculating the pairwise information between chords. There are many more sources available from local features such as normals, tangents up to richer features such as line profiles [192], Haar-like features [202], and so on. First, distances are explored as the absolute distance between any two boundary points is converted to relative measurement using global information. Second, angular measurements describe the relative spatial arrangement between two points on the boundary. However, as this requires a third point to allow determining an angle, we propose several of third point selection procedures. Figure 3.6 illustrates the information collected from the two sources. It shows the measurements from (a) the distance to determine the ratio of chord lengths, and (b) the angles between a pair of chords.

3.4.5.1 Distances

One source of information for the description is distance between two points P_i and P_j on the boundary. This descriptor is on the distance d_{ij} which describe the length of the chord between the boundary points. The length d_{ij} is calculated, which is defined as the Euclidean distance between the coordinates of the boundary point tuple (P_i, P_j) as



(a) Distance (b) Angle

Figure 3.6: Our Structured Measurement Descriptors use the information of a pair of chords to derive invariant information about shape. We measure distances (a) to determine the ratio of chord lengths, and angles (b) of enclosed by the chords.

$$d_{ij} = \|\overline{P_i P_j}\| = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}, \tag{3.10}$$

where the points are $P_i = P(x_i, y_i)$ and $P_j = P(x_j, y_j)$. Such distance measurements are invariant to translation and rotation. One can use a normalization to achieve scale invariance. For example, the median distance between all combinations of the boundary points P_i and P_j may be used to robustly normalize the other distances, resulting in invariance to similarity transformation. The resulting descriptor has values in the range between 0 and 1. However, our goal is to define pairwise chord interactions, which measure the ratio of length two chords (P_i, P_j) and (P_i, P_k). We measure the relative difference of their lengths by

$$\gamma_{ij} = \frac{\|\overline{P_i P_j}\|}{\|\overline{P_i P_k}\|}, \tag{3.11}$$

where $\| \dots \|$ denotes the length of a chord line as defined above, and $\overline{P_i P_j}$ define a chord and P_k is a third point used to define a second chord $\overline{P_i P_k}$.

3.4.5.2 Angles

The second source of information are angular measurements. The descriptor based on angles α_{ij} describe the relative spatial arrangement of the sampled points. An angle α_{ij}

is calculated between a chord $\overline{P_iP_j}$ from a index point P_i to another sampled point P_j and a chord $\overline{P_iP_k}$ from P_i to P_k by

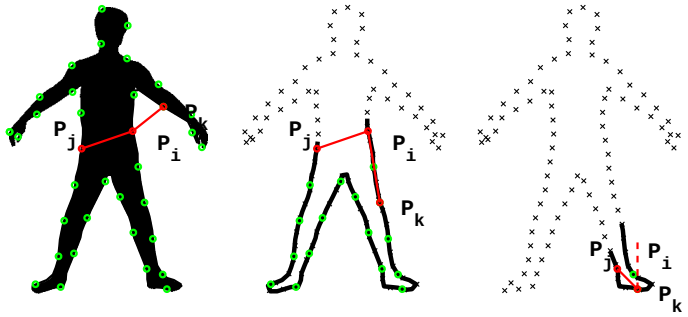
$$\alpha_{ij} = \sphericalangle\left(\overline{P_iP_j}, \overline{P_iP_k}\right), \tag{3.12}$$

where $\sphericalangle(\ldots)$ denotes the angle in the range of 0 to Pi (later normalized to 1) between the two chords and a third point P_k which defines the second chord. The selection of the third point is crucial and defines different properties on the description, which we will discuss in the following section. However, since relative angles are preserved by a similarity transformation, this description is invariant to translation, rotation and scale.

3.5 Region, Contour and Fragment Descriptors

The *Structural Measurement Descriptor* is designed to allow more flexible part description yet discriminative description for the consequent matching of parts. In the following we discuss the characteristics of our description and show specific formulations for capturing the essence of their underlying extent of shape in terms of closed regions, contours or local fragments.

As defined in our notation, a region is a closed boundaries around an object of interest. This property defined a circular ordering of the points, where the first point is also the last point. It allows to capture the connections between close by points. For contours, the definition states that they define an open sequence of points anywhere on the boundary of the object. There is no knowledge about where the end points lie and only that they are not the same. The property of contours is that they may be of arbitrary length, which is also not known in advance. Fragments are defined as finite sequences of boundary points, with a known and equal length. The fragments capture the smaller amount of information of a shape as their extent is limited. Each of these types of boundaries defines different characteristics, which we capture by a different selection of the pairs of chords. Figure 3.7 gives an overview of the types of boundaries and their chord selection methods. Figure 3.8 illustrates the descriptors for these chord selection methods for different shape primitives, and Figure 3.9 for different scales of the shape shape primitive.



(a) Region (b) Contour (c) Fragment

Figure 3.7: Structured measurement descriptors use different selection of chord pairs for different types of boundaries: regions, contours, fragment description.

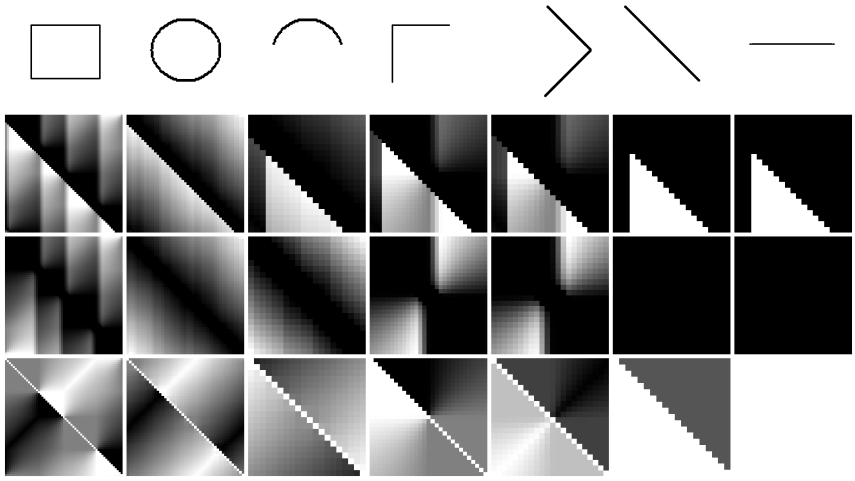


Figure 3.8: Visualization of the Structural Measurement Descriptors for selected a set of primitives. Second row shows the region descriptors. Third row shows the contour descriptors. Note how each part of primitive is included in its respective closed primitive (square and circle), except for the last row which the fragment descriptors.

3.5.1 SMD for Region Description

Regions form closed boundaries around an object of interest. This means that the sequence of boundary point is circular and may have an arbitrary starting point. This property is used when matching closed shapes as discussed in Chapter 4. For description, we are interested in the circularity of region boundaries points, as it allows us to build a description in the following fashion. For example, an angle α_{ij} is calculated be-

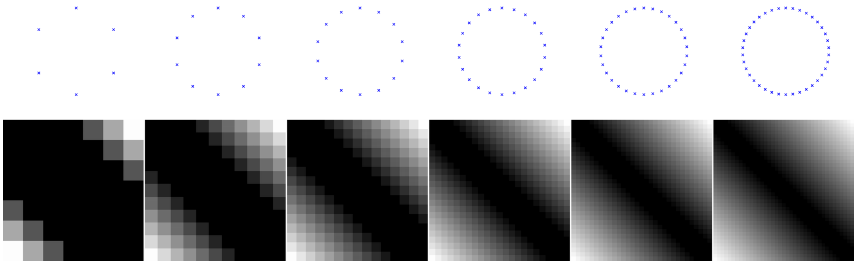


Figure 3.9: Scale is implicitly included in the Structured Measurement Description. The further measurements are apart, the more globally they capture the underlying shape.

tween a chord $\overline{P_iP_j}$ from a reference point P_i to another sampled point P_j and a chord $\overline{P_iP_{j-\Delta}}$ from P_i to $P_{j-\Delta}$ by

$$\alpha_{ij} = \sphericalangle\left(\overline{P_iP_j}, \overline{P_iP_{j-\Delta}}\right), \tag{3.13}$$

where P_i and P_j are the i^{th} and j^{th} points in the sequence of sampled points of the contour and Δ is an offset parameter of the descriptor. The point $P_{j-\Delta}$ is Δ positions before P_j in the sequence. As the sequence is circular and the first and last points overlap, this captures the essence of a region descriptor, which follows the circular sequence along the shape back to the starting point.

3.5.2 SMD for Contour Description

Contours are parts of a boundary of an object, can be arbitrarily long and may contain occlusions as discussed in Chapter 5 For the contours we build a description as following. The measurement for an angle α_{ij} is calculated between the two chords $\overline{P_iP_k}$ and $\overline{P_jP_k}$ and P_k is again Δ positions away as defined by

$$\alpha_{ij} = \begin{cases} \sphericalangle(\overline{P_iP_j}, \overline{P_iP_{j-\Delta}}) & \text{if } i < j \\ \sphericalangle(\overline{P_iP_j}, \overline{P_iP_{j+\Delta}}) & \text{if } i > j \\ 0 & \text{if } abs(i-j) \leq \Delta \end{cases} \tag{3.14}$$

where P_k is the third point and is chosen depending on the position of the other two points to ensure that the selected point is always inside the contour. This allows us to further enforce the self-containment of the description of any of its parts. It is important for contours as the source of information for the boundary stems from from underly-

ing image edges. Hence the boundary of the shape is often broken up into parts or contaminated with noise and occlusion contours.

3.5.3 SMD for Rotation-Variant Description

The contour descriptions are designed for rotation invariance. However, often rotation-invariance is not required and the additional invariance would leads to less discrimination. Thus the orientation should be separately encoded to not lose any additional information [67]. This is especially valuable when the contour parts only contain limited information about the local structure. Hence, for contours we can build a description in the following fashion. An angle α_{ij} is calculated between a chord $\overline{P_iP_j}$ from a reference point P_i to another sampled point P_j and a chord $\overline{P_iP_\infty}$ from P_i to P_∞ by

$$\alpha_{ij} = \sphericalangle\left(\overline{P_iP_j}, \overline{P_iP_\infty}\right), \tag{3.15}$$

where P_∞ is the point at infinity, which results in a rotation-dependent description. For some tasks the orientation is known beforehand or is explicitly desired, for example when drawing the sketch of a object by hand or analyzing subsequent video frames.

3.5.4 SMD for Fragment Description

Fragments are very small parts of a contour and only deal with a very limited spatial extent. This brings various benefits, however also reduces the information that is captured by such local structures. The spatial extent now determines the size of the part. Contrary to a contour description, only a subset of points are available to derive a fragment description. The support window is given by the number of boundary points surrounding the fragment in each direction. Additionally, since the extent is limited we can infer a clockwise sorting of the boundary points. Thus the points are not only in a sequence, yet also the representation is now invariant towards the direction of the sequence.

For fragments it is especially valuable to maintain as much of the information as possible due to their limited local structure. Hence, we build a description in the following fashion encoding its orientation using a reference point on its bounding rectangle. An angle α_{ij} is calculated between a chord $\overline{P_iP_j}$ from a reference point P_i to another sampled point P_j and a chord $\overline{P_iP_*}$ from P_i to P_*, which is defined by the upper left corner of the bounding box of a fragment. In such a way, we define

$$\alpha_{ij} = \sphericalangle(\overline{P_iP_j}, \overline{P_iP_*}), \tag{3.16}$$

where P_* is a fixed reference point relative to the fragment. It is defined by the upper left corner of the bounding box of the boundary points for this fragment. This description encodes the limited spatial environment of each fragment. This limited support window leads to a compacter description as less points ($K \ll N$) are included compared to contour description. Partial description is achieved through redundant overlapping of such fragments. In this way a fixed length of K points results in a fixed descriptor of K^2, which can be used for machine learning as shown in Chapter 6.

3.6 Experimental Evaluation

Due to the large variety of shape-based descriptors, it is difficult to express the individual benefits and the best use. Most importantly, a direct comparison for region and contour matching is not possible because one requires partial matching. This is discussed and evaluated in the next chapters in detail. In this section, we compare the distinctiveness of local fragment description, which have only a local support window. Therefore we design an evaluation looking at the classification rate of our structural shape description compared to related work for local interest point description.

In this experiment, we evaluate different shape descriptors within different algorithms for a classification task. In particular, we compare our proposed structural descriptors for regions, contours and fragments to three other descriptors from related work, beam angle (BA) [10], turning angle (TA) [37] and shape context (SC) [19]. The classification algorithms we used were a random forest, linear SVM and a simple nearest neighbor classifier.

In the experimental setup we classify individual edge pixels on the detected edges in the test images into foreground or background. Specifically, we compare the per-pixel classification results to the ground-truth annotations. Hence, we define a classification error, describing the ratio of false classifications to all edge pixels. The protocol for the experiment uses 50% of the images for learning a classifier and 50% for testing. We extracted fixed-length fragments (or quadratic patches containing edges for SC) at varying sizes from the images, such that for all images a reasonable number of foreground edges remained in the test set. For data we use an object class from the ETHZ Shape dataset [66], which expresses the highest intra-class variability, namely the giraffe class.

Length	Region	BA [10]	Contour	TA [37]	SC [19]	Fragment
l=51	41.67	41.61	41.13	40.49	37.13	33.60
l=41	43.15	42.67	42.68	42.14	37.51	35.93
l=31	43.43	43.48	42.99	42.81	38.33	38.58

Table 3.2: Classification error of shape fragment descriptors at several lengths, showing the per-pixel classification error in % (see text for definition) when learned in a random forest classifier. Our fragment descriptor yields the lowest classification error of 33.60% for a length of 51 points, significantly lower than previous state-of-the-art shape descriptors.

In Table 3.2 we list some results for fragment lengths / patch sizes of 31, 41 and 51 pixels, when learning the individual descriptors in a random forest framework. As shown, the longer the fragments, the better the discrimination is. The fragment descriptor outperforms the other descriptors at these very limited local support windows. It achieves its best performance at length $N = 51$ and hence, it is the best suited setup for use in a discriminative setting. Please note that increasing the length even more may result in better classification scores, however, the number of edges belonging to the object category decreases. Many extracted edges have a smaller length and the best tradeoff between best classification and maximum recall.

We also evaluate different sampling strategies for selecting the individual points on the fragments and found that there was no significant change in the classification performance. Specifically, our descriptor does not show a drop in performance for different sampling distances between one and ten pixels. However, this comes with a positive side effect in terms of speedup due to the reduced data and quadratic descriptor dimensions, which are used for processing. Changing the sampling density for the patch based descriptors reduced the data and hence the robustness of the statistical description, and we noticed slight drops in performance.

Using linear support vector machine (SVM) for classification results in approximately similar relative distributions of the scores. However, the absolute mean error is on average about 5% to 10% higher, which suggests that random forest classifiers are better suited for our task for two reasons. First, the random forest classifier is able to handle non-linear data implicitly, where as the linear SVM on linear decision boundaries. For the SVM, one would need to design a non-linear kernel mapping for this shape description to improve the results [63, 134]. Second, the random forest is known to better cope with label noise and high-dimensional descriptors. Using a nearest neigh-

bor search, the performance is much slower yet identical, as it seems that enough data is available to always find the best match.

3.7 Conclusion

In this chapter we proposed a novel description of shape denoted as Structured Measurement Descriptors (SMD) to capture the perceived holistic geometric layout of points of an object. The description is inspired by the psychological principles of good continuation of Gestalt and collinearity of Biedermann. Further, the design of the descriptor captures local as well as global information about a shape. Each subpart is individually described and also captures the configuration of parts in a holistic manner. This allows partial description and matching for handling deformations caused by noise, articulation, or occlusion.

The underlying representation is a sequence of boundary points, which may come a closed region, an open contour from edge images or as small fragments describing a local support window. Our novel formulation of Structured Measurement Descriptors generalizes these sources of boundaries and description. However, each level of shape abstraction has its own specific method for use in retrieval, classification, and detection tasks. Region description captures the full scale of an object and incorporates the most information compared to the other types.

In Chapter 4 we show how a partial similarity measure for regions is used to perform retrieval and clustering of closed regions. The contour description has an implicit unknown scale, however partial matching to a reference template allows to find alignment and scale though the correspondence. In Chapter 5 we show how partial contour matches can be used for efficiently locating objects in cluttered images. The fragment description carries the least information from the limited local support window, however we show how this local window allows to exploit the sequence ordering for a known direction and fixed-length feature description. In Chapter 6 we show how such local shape fragments are used to efficiently train a robust category model by jointly optimizing the classification and constellation of shape fragments.

The whole is other from the sum of its parts.

Kurt Koffka

4

Partial Similarity for Region Matching

In this chapter we propose a novel method for matching the shape of regions, which has applications in many computer vision tasks. Related work deals with robust distance functions or employs graphical models to infer coherent structure when matching local interest points, as reviewed in Section 4.2. In contrast, we show that the *Structural Measurement Descriptor* (SMD) is well-suited for enforcing coherent structure. The hierarchical description encodes all parts into one coherent descriptor, where the spatial extent of parts is not known, which makes this approach incompatible with standard matching or detection frameworks. Our solution is a 3D tensor containing all combinations for correspondence, length and similarity. In this chapter we focus on matching closed regions given by a binary mask and propose a novel method for measuring a balanced partial similarity, as described in Section 4.3. In the experimental evaluation our partial similarity measure achieves superior computational efficiency of only 3 ms per match as well as competitive results to state of the art in shape retrieval and clustering on the well-known KIMIA-25, KIMIA-99 and MPEG-7 silhouette datasets, as shown in Section 4.4. For the task of object detection in cluttered images, however, another method is required for extracting partial matches and is shown in the next chapter.

Contents

4.1 Introduction

Shape matching is a well investigated problem in computer vision and has versatile applications, for example in shape retrieval [19, 173, 180, 213], object detection [57, 66, 153], image retrieval [146], object tracking [203], or action recognition [204]. Most of these tasks require a shape similarity measure between shapes and a way to cope with the occlusions and deformations. The large field of related work deals with these problems in two ways. First, distance functions are designed for robust comparison of shapes in terms of statistical distribution of global point sets capturing coarse structures, or edit distances in shock graphs to explain topological changes. Second, deformation costs and graphical models are introduced to infer coherent structure during matching of semi-local interest points.

In this chapter we follow a different strategy, namely that of partial matching of a hierarchical description of shape. The shape description introduced in the previous chapter is designed for exactly this purpose. The Structural Measurement Descriptor is a holistic description allowing each part of a shape to be described and matched independently as well as jointly. However, this unique property is not compatible with standard matching or detection frameworks. The principle difference is that the Structural Measurement Descriptor is a single coherent descriptor for the entire shape, which is hierarchical and each part may be of arbitrary spatial extent. This is contrary to most related work, which uses known spatial extents in their global or local shape descriptions. For closed regions most approaches use global descriptors and distance functions which are robust to occlusions. For cluttered images local descriptions are used with a focus on coping with clutter and occlusions.

A partial shape matching algorithm has to fulfill several conflicting requirements. First, the method has to provide a distance function to measure the similarity of two

shapes, returning a single similarity value as result. Second, it has to be able to retrieve partial results between similar parts of the two shapes. Third, it has to provide correspondences between the matching parts to align the matching and complete the missing part of the shape. Finally, the algorithm has to be efficient due to the large number of possible matching combinations. This chapter deals with the structure in matching shapes. We propose a novel method for matching the structure of shapes by exploiting their sequence information. This sequence is implicitly encoded in the Structural Measurement Descriptor and hence can be further exploited to extract structurally coherent partial matches.

The main aspect is the design of a partial similarity distance and its efficient calculation. Such a distance deals with the balance between partiality and similarity while also considering the structure of the shapes. We build a 3D tensor containing all combinations for matching two shapes. This tensor calculation however is generic to allow any content to be matched. It does not necessarily have to be shape information, as long as the sequential ordering requirement is fulfilled. Our matching method focuses on the order-preservation and pairwise geometric properties inherent in the Structural Measurement Descriptor. We define a global optimal Pareto frontier to calculate a similarity measure between shapes balancing the partiality and similarity of two shapes.

In this chapter we focus on a novel method for measuring partial shape of closed regions and show vast performance improvements over state of the art. The method is two orders of magnitude faster than comparable work, requiring only a 3 milliseconds per match. Experimental validation in tasks of shape clustering and shape retrieval on standard shape datasets like KIMIA-25, KIMIA-99 and MPEG-7 show that we achieve state-of-the-art results at greatly reduced computational costs. In the following chapter we show how to use the 3D similarity tensor to extract valid partial matches, which are used for category object detection in cluttered images.

4.2 Related Work

There exists a long history in related work for shape matching. Considering the ways previous work deals with coping with noise, occlusions and defining a similarity measure for the matched parts, there are two approaches. First, distance function are designed for robust comparison of the data points distribution. Second, deformation costs in graphical models are introduced to infer coherent structure during matching of (semi-local interest) points. These two ideas relate to measuring the energy for aligning the

data, or measuring the difference between the data and its alignment. The latter follows in spirit the idea of Kendall's theory of shape [102], which defines a shape as what remains after normalization, which removed deformations by similarity transformations.

4.2.1 Distance measures

A distance function measures the similarity between two given sets of data, for example, as feature vectors, in color space, as point sets or closed regions of shape. Since partial matching requires a form of measuring the quality of the correspondence between shapes, one needs a distance function which captures the similarity of two shapes robust to occlusions and noise. In this section we will review some of these distance functions.

The most direct way is to determine a similarity between individual points. The general form is denoted as the Minkowski distance [142] and defined as

$$Lp(a,b) = (\sum_{i=0}^{d} = ||a_i - b_i||_p)^{\frac{1}{p}} \tag{4.1}$$

where a_i is the i^{th} dimension of the point. For specific cases of p, this simplifies to the well-known Euclidean distance (L2-norm, p=2) or Manhattan distances (L1-norm, p=1). It gives a measure how far apart two points are. For a set of points this becomes more difficult, as we need to handle outliers and the assignment between points.

Robust distance functions are based on the principle idea that only the minimum distance is sufficient to measure shape similarity. The Bottleneck distance [56] is defined as the minimum over the largest distance for all combination of two points. The Hausdorff distance [97] is defined as the largest minimum distance over the point sets (and is sensitive to outliers and ignores order of points). Fréchet distance [3] is defined as smallest distance over all combinations of two points while preserving the adjacency of points along a curve, employing the metaphor or a man and a dog on a leash to measure how far apart the points are allowed to be.

The Procrustes shape analysis [84] uses Euclidean distance between two sets of points. The two sets of points are overlayed and a distance between the superimposed points determines the quality of the shape alignment, refer to Equation 3.2.

For statistical histograms as available in chord distributions by [41, 155], the Earth Mover's Distance (EMD) [167] is a well-suited distance, which measures the amount of changes required between the discrete histogram bins to find the minimal distance. Recently faster methods for Earth Mover's Distance have been proposed in [88, 128].

Chamfer matching [16] is a popular method for comparing two shapes in terms of images. The chamfer distance measures the average distance from a set of points to its nearest neighbors. It is commonly used for matching in edge images rather than the comparing shapes. However, it is sensitive to clutter [129, 193], yet its superiority in computational performance makes it popular. There have been numerous extensions including hierarchy [30], orientation [183] and direction [129]. Still it is based on a rigid template and requires explicit search of rotation and scale.

Edit distances stem from the field of string sequence matching and forms a class of distance measures which count the number of operations required to transform a string into another, deleting, inserting, or skipping string elements. Shock graphs [186] view the evolution of such Medial Axes transforms [25] during their formation, and describe the singularities in a graph representation. The graph is represented as a tree and the edit events (adding or removing nodes) are defined as the similarity measure.

4.2.2 Deformation measures

The second approach is measuring the deformation costs for transforming one shape onto another. This often requires an initial set of correspondences to calculate the costs and infer a subsequent transformation update.

One example is the Thin Plate Spline (TPS) transform [54], which has been used in non-rigid matching method for aligning two point sets [38]. It measures the amount of bending energy required to fit one shape onto another. Both require identified point sets which have to be aligned, either by cleanly segmenting the points from clutter or providing correspondences between them.

Similar, in [178] and [17] the deformation energy is measured as the minimal effort needed to transform one contour into the other. [116, 118] define their similarity function by only allowing a simplification of the shape to occur. Further they propose not to measure the cost of deforming a shape, however instead the shape similarity after deformation has been carried out, similar to aligning after Procrustes analysis.

The more general correspondence assignment is a optimization problem in many fields. The assignment between two sets of points consists of a weighted graph. The graph connects each point combination of the two sets with an edge, which has a weight corresponding to the cost of assigning these two points. Depending on the knowledge available, this graph may be reduced to a bipartite graph or even a sequence of points. [20] uses a quadratic assignment problem to align two sets of interest points. [18] uses a

linear assignment solver known as the Hungarian algorithm [110] for aligning the point sets. This has been extended in [175] by defining it as an cyclic order-preserving assignment problem (COPAP), which is essentially matching the point in a sequence. Further variants include the cyclic dynamic time warping solution by [33] and the dynamic programming for aligning sequences by [58].

4.3 Structure in Partial Matching

In this chapter, we propose an alterative to the coarse statistical distance function and time intensive correspondence assignment via graph matching. The most important part of designing a shape matcher is the choice of the shape representation which has a significant effect on the matching step. We directly exploit the properties included in the Structural Measurement Descriptors (SMD). First, the description is a hierarchy of the entire shape, which includes all part descriptions and configurations. The nature of this full description does not limit the spatial extent. In comparison, local interest points always have a limited local support window, which may limit the distinctiveness of the local description. In the Structural Measurement Descriptor the matching decides the spatial extent. Second, the description contains the ordering sequence due to the ordered measurements stored in the structured description.

For example, the Shape Context [19] descriptor loses all the ordering information due to the histogram quantization and for that reason does not consider the influence of the local neighborhood on single point matches. This has to be recovered by solving the expensive assignment problem, even for the special case of the order-preserving assignment problem [175]. The main challenges in this field lie in the implicit handling of occlusions, measuring the partiality of a match and reducing the computation of the solutions. For example, the method in [175] is lacking the ability to handle occlusions. Figure 4.1 illustrates a result of our proposed method, where our partial matching enables to handle occlusion.

We introduce a novel shape matching method denoted as *IS-Match* (Integral Shape Match). The matching exploits the structure and sequence information given by an ordered sequence of points on a boundary, which is encoded in the Structural Measurement Descriptors (SMD). In this way, there is no need for expensive graph matching, since the structure is already encoded in the description. The proposed method is divided into the three following parts:

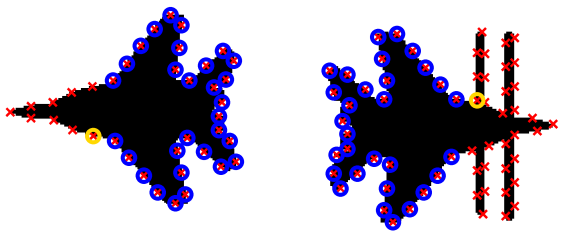


Figure 4.1: Result of our partial shape matching where sampled points are indicated by the crosses, and matching results are shown with circles.

- **The Structural Measurement Description** (SMD) is used for a region specific shape descriptor. The closed region information enables specific properties such as known scale and cyclic sequence which allow to match across the starting and end points of a region.

- **The 3D Similarity Tensor** compares the descriptor matrices of two shapes by an efficient integral image-based algorithm. This allows calculation of the similarity for any subparts of the shapes with an runtime complexity of $O(n^2)$.

- **The Pareto-Optimal Partial Similarity** analysis the partial similarities on its global optimal Pareto frontier and calculates the corresponding Salukwadze distance as a measure between shapes optimally balancing between partiality and similarity.

In this chapter, we focus on the problem of matching the shapes of closed regions, which enables the region-specific shape description. However, it requires a cyclic search along the region boundary points and a measure to evaluate the partiality as well as the distance between shapes. The 3D similarity tensor is a general method for matching two Structural Measurement Descriptors, and as shown in the next Chapter it is used for partial matching of contours using a different matching strategy.

In Section 4.3.1 we describe the shape description for closed regions. In Section 4.3.2 we propose a method for efficient calculation of all combinations in correspondence and lengths the matching of two shapes which is stored in a 3D similarity tensor. In Section 4.3.3 we show how to derive an optimal balance between partiality and similarity for measuring shape similarity. In Section 4.3.4 we analyze the required computational complexity for a matching of two shapes.

4.3.1 Region Description

As we focus on closed regions in this chapter, our input data for shapes are either binary masks containing a single region or an otherwise extracted sequence of boundary points $\mathcal{R} = \{P_1, P_2, \ldots, P_M\}$, where the boundary is closed, i. e. P_1 and P_M are the identical points. We use these boundary points as a representation and exploit the ordering of the points to formulate the description and the matching to preserve the order. We use the Structural Measurement Descriptors (SMD) introduced the previous chapter to derive a description specific for closed regions. For description, we are interested in the circularity of region boundaries points, as it allows us to build a description in the following fashion. An angle α_{ij} is calculated between a chord $\overline{P_iP_j}$ from a reference point P_i to another sampled point P_j and a chord $\overline{P_iP_{j-\Delta}}$ from P_i to $P_{j-\Delta}$ by

$$\alpha_{ij} = \sphericalangle\left(\overline{P_iP_j}, \overline{P_iP_{j-\Delta}}\right), \tag{4.2}$$

where P_i and P_j are the i^{th} and j^{th} points in the sequence of sampled points of the contour and Δ is an offset parameter of the descriptor. The point $P_{j-\Delta}$ is Δ positions before P_j in the sequence. As the sequence is circular and the first and last points overlap, this cyclic sequence captures the essence of a region descriptor.

Since the description depends on which point is chosen as the starting point of the sequence. Given a region and its closed boundaries around an object of interest, the starting point is also the ending point. Yet the starting point may be any point along the region and the descriptor depends on which point is chosen as the first point of the sequence. For example the descriptor matrix Ψ changes to

$$\Psi_{(k)} = \begin{pmatrix} \psi_{kk} & \cdots & \psi_{k1} & \cdots & \psi_{k(k-1)} \\ \vdots & \ddots & \vdots & \iddots & \vdots \\ \psi_{1k} & \cdots & \psi_{11} & \cdots & \psi_{1(k-1)} \\ \vdots & \iddots & \vdots & \ddots & \vdots \\ \psi_{(k-1)k} & \cdots & \psi_{(k-1)1} & \cdots & \psi_{(k-1)(k-1)} \end{pmatrix} \tag{4.3}$$

if the k^{th} point is set as the first point of the sequence. For closed boundaries, these two matrices $\Psi_{(k)}$ and A are directly related by a circular shift. Matrix Ψ can be obtained by shifting the $\Psi(k)$ matrix $k-1$ rows up and $k-1$ columns to the left. This is an important property for the efficient descriptor matching. Note this property is only valid for closed regions, as open contours do not require to set a fixed starting point.

4.3.2 Partial Similarity Tensor

The Structural Measurement Descriptor (SMD) for a shape is single coherent hierarchical description of all parts and configurations. In this section we describe how to extract similarities for all these parts. Matching a part requires finding the correspondence between the reference and query descriptor. The description for shapes constitute the basis for matching a point of the reference shape to a point of the query shape. Since the structure is implicitly order-preserving, thus is the consequent matching formulation. To find a partial match between two given shape regions R_1 and R_2 the corresponding descriptor matrices Ψ_1 with size $M \times M$ and Ψ_2 with size $N \times N$ are compared. For notational simplicity we assume that $M \leq N$.

The aim of partial shape matching is to identify parts of the two shapes which are similar to each other. In terms of comparing the two descriptor matrices this equals to finding $l \times l$ sized blocks starting at the main diagonal elements $\Psi_1(r,r)$ and $\Psi_2(q,q)$ of the two descriptors which yield a small average difference value $\Delta(r,q,l)$ defined by

$$\Delta(r,t,l) = \frac{1}{l^2} \sum_{i=0}^{l-1} \sum_{j=0}^{l-1} \| \Psi_1 \left(r+i, r+j\right) - \Psi_2 \left(q+i, q+j\right) |_2 \tag{4.4}$$

where $|...|_2$ is the L2-norm between the blocks. This is possible due to the previously explained property that a different starting point leads to a circular shift of the descriptor matrix, see Equation 4.3. To find the matching blocks, all different matching possibilities for correspondence and length have to be considered and the brute-force method becomes inefficient for larger number of points. A closer look into Equation 4.4 shows that three different loops for the variables q, r and l have to be handled.

- **Changing the reference starting point (r):** There are $N \geq M$ possibilities to match the two boundaries onto each other by rotating the shape R_1 with the lower number of sampled points M within the shape R_2.

- **Changing the query starting point (q):** Once a reference assignment between the two shapes is fixed by the previously described loop, the starting point for the query boundary must be chosen. There are again $M \leq N$ possibilities to choose this starting point.

- **Adapting sequence length (l):** Once the starting points are fixed, the length l of the sequence can be adapted, which is the most challenging part.

This presents a high combinatory complexity, therefore a high efficiency of this step is important. The adaptive length makes the matching unique and challenging. In contrast to quadratic assignment problems where all combinations of two points are considered, the number of points considered in matching is variable.

We propose an algorithmic optimization to overcome the limitations of the sole brute-force approach, which is based on an adaption of the Summed-Area-Table (SAT) [44] or more popularly known as integral images [202] to calculate all the descriptor differences $\Delta(q, r, l)$. The integral image approach allows to calculate the value of rectangle features, for example, the sum of all pixel values for any size and any location in constant time. The approach works by precalculating the values according to an integral function, which combines the results of all smaller rectangles into one value. When an arbitrary location is evaluated, only four constant lookup operations for adding and subtracting are performed to determine the value.

We use this concept to precalculate the differences between the Structural Measurement Descriptors. Hence, we calculate integral difference matrices, which store the difference between two descriptors. For calculating the similarity scores for all possible configuration triplets $\{q, r, l\}$ in the most efficient way N integral images $Int^1 \dots Int^N$ each of size $M \times M$ are built for N descriptor difference matrices Δ^n defined by

$$\Delta^n = \Psi_1 (1 : M, 1 : M) - \Psi_2 (n : n + M - 1, n : n + M - 1) , \tag{4.5}$$

where $\Psi_1 (1 : M, 1 : M)$ is the cropped $M \times M$ square sub-block of Ψ_1 including all elements of the first to the M^{th} row and the first to the M^{th} column. The difference matrices Δ^n represent the N possibilities to match the boundaries onto each other. Based on these N integral images $Int^1 \dots Int^N$ the difference values $\Theta(r, q, l)$ can be calculated for every block of any size starting at any point on the diagonal in constant time, which reduces the computational complexity. The resulting triplets $\{r, q, l\}$ provide a similarity measure for every starting point in the reference, query, as well as every length.

4.3.3 Region Shape Similarity

Having defined all similarities between the correspondences and length, we are one step closer to defining the similarity between the shape of closed regions. It is important to provide a reasonable similarity measure in addition to the identified matching point sequences, e. g. for tasks like shape retrieval. Commonly, a combination of descriptor difference, matched shape distances like the Procrustes distance [84] and bending en-

ergy of an estimated transformation like a Thin Plate Spline (TPS) [54] given a set of correspondences is used. However, since we are analyzing more than just the statistical similarity of two shapes, we seek an assignment between all the points on the reference shape onto the query shape. This assignment however requires the knowledge of occlusions and partial similarity, which is not compatible with a pure transformation-based similarity estimation calculated of the complete shapes.

The straight-forward way is to select the best matching correspondence by selecting the lowest distance value from the 3D tensor. However, this will result in a trivial solution of minimal length. Therefore we focus on partial similarity evaluation, and we define a partial similarity measure over the 3D similarity tensor inspired by Bronstein et al. [34]. The Pareto frontier for quantitative interpretation of partial similarity defines the two quantities:

- **Partiality** $\lambda\left(X', Y'\right)$, which describes the length of the parts in terms of percentage covered of the entire region boundary between 0 and 1 (the higher the value, the larger the partiality and hence the smaller the part).
- **Dissimilarity** $\varepsilon\left(X', Y'\right)$, which measures the dissimilarity between the corresponding parts, where X' and Y' are the two parts of the shape (the smaller the value, the more similar the parts).

A pair $\Phi(X^*, Y^*) = \left(\lambda\left(X^*, Y^*\right), \varepsilon\left(X^*, Y^*\right)\right)$ of partiality and dissimilarity values, fulfilling the criterion of lowest dissimilarity for the given partiality, defines a Pareto optimum. All Pareto optimums can be visualized as a curve, and are referred to as the set-valued Pareto frontier.

Since finding the Pareto frontier is a combinatorial problem in the discrete case, mostly rough approximations are used as final distance measure. Our matching algorithm automatically evaluates all possible matches for all possible lengths resulting in the 3D tensor similarity. Therefore, by focusing on the discretization defined by our point sampling and the property that we are dealing with closed regions, we can calculate a global optimal Pareto frontier, by returning the minimum descriptor difference for all partialities. Thus for every part in the reference region we can infer the best similarity and partiality criteria in the parts of the query region.

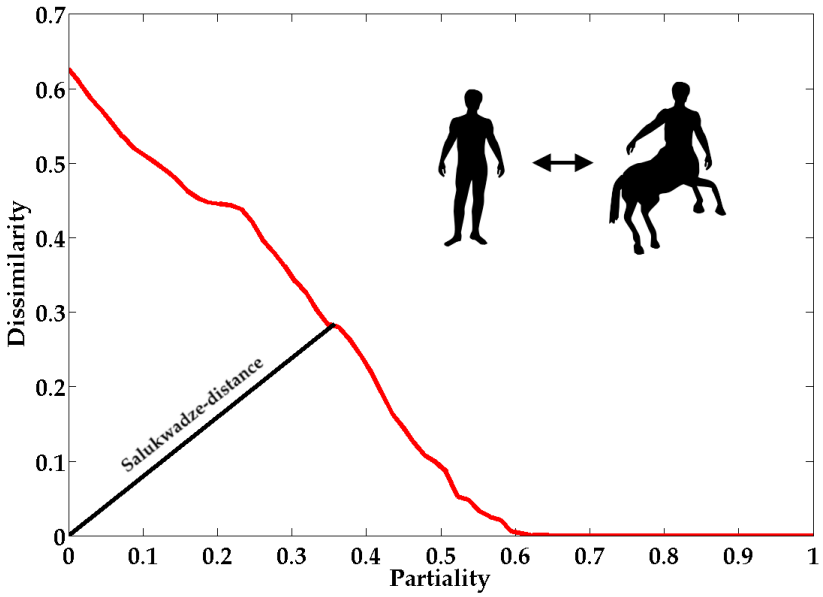


Figure 4.2: Our shape matching algorithm *IS-Match* analysis all possible correspondences for all sequence lengths which allows calculating a global optimal Pareto frontier. The Salukwadze distance is returned as similarity score balancing the partiality and similarity of the match.

Finally, we determine a single value measuring the overall similarity between two provided shapes to allow comparison using the Salukwadze distance [170]. This distance $\mathcal{D}_{Salukwadze}$ is calculated based on the Pareto frontier by

$$\mathcal{D}_{Salukwadze} = \inf_{(X^*,Y^*)} |\Phi(X^*, Y^*)|_1 , \tag{4.6}$$

where $|...|_1$ is the L1-norm of the vector. Therefore, $\mathcal{D}_{Salukwadze}(X, Y)$ measures the distance from the utopia $(0, 0)$ to the closest point on the Pareto frontier. The Salukwadze distance now delivers a similarity score when matching between the shapes of two closed regions. Figure 4.2 illustrates the calculation of the global optimal Pareto frontier and Salukwadze distance in the example of a centaur and a human.

4.3.4 Computational Complexity

Our integral image based matching algorithm detects partial matches with lower computational complexity. The method returns the set of partial matches by evaluating a global optimal Pareto frontier to define a partial similarity measure between shapes.

Method	N	Complexity	Runtime
Sebastian et al. [178]	-	-	1000 ms
Tu and Yuille [197]	-	-	200 ms
Felzenszwalb et al. [61]	100	$O(t^3k^3)$	500 ms
Scott and Nowak [175]	100	$O(mnl)$	450 ms
Ling and Jacobs [127]	100	$O(t^2n)$	310 ms
Belongie et al. [19]	100	$O(t^2n)$	200 ms
Schmidt et al. [173]	200	$O(t^2log(t))$	-
Brendel and Todorovic [33]	100	$O(nm)$	200 ms
Our work	**100**	$O(nm)$	**120 ms**
Our work	**30**	$O(nm)$	**3 ms**

Table 4.1: Comparison of computational complexity and runtime in milliseconds for a single match for best performance. For typical 100 points we require the smallest runtime. Yet we our algorithm only requires 30 points to achieve competitive results on reference datasets. Hence, an additional significant speedup of runtime is achieved!

These partial matches therefore allow alignment between differently articulated and occluded shapes.

An exhaustive search over all possible matches for all possible lengths has a complexity of $O(2^{n+m})$. Our proposed approach based on integral image analysis enables matching in $O(nm)$ time, where n and m are the number of sampled points on the two input shapes.

For comparison, Table 4.1 summarizes complexities and runtime of current state-of-the-art shape matching methods. As it is shown in Section 4.4, only 30 sampled points are required to provide close to state-of-the-art shape retrieval results, which is possible within only $3\,ms$. Please note, that the runtime may vary due to differences in implementations and machine configurations. But as can be seen in general *IS-Match* outperforms state-of-the-art concerning computational complexity and actual runtime. To the best of our knowledge this constitutes the fastest method for combinatorial matching of 2D shapes published so far.

4.4 Experimental Evaluation

In the experimental evaluation we analyze the overall quality of our partial shape matching method *IS-Match* on three common datasets. We use the KIMIA-25 and KIMIA-99 from Sharvit et al. [180] and further evaluate on the largest and currently most important benchmark for evaluating shape matching algorithms, the MPEG-7 silhouette dataset

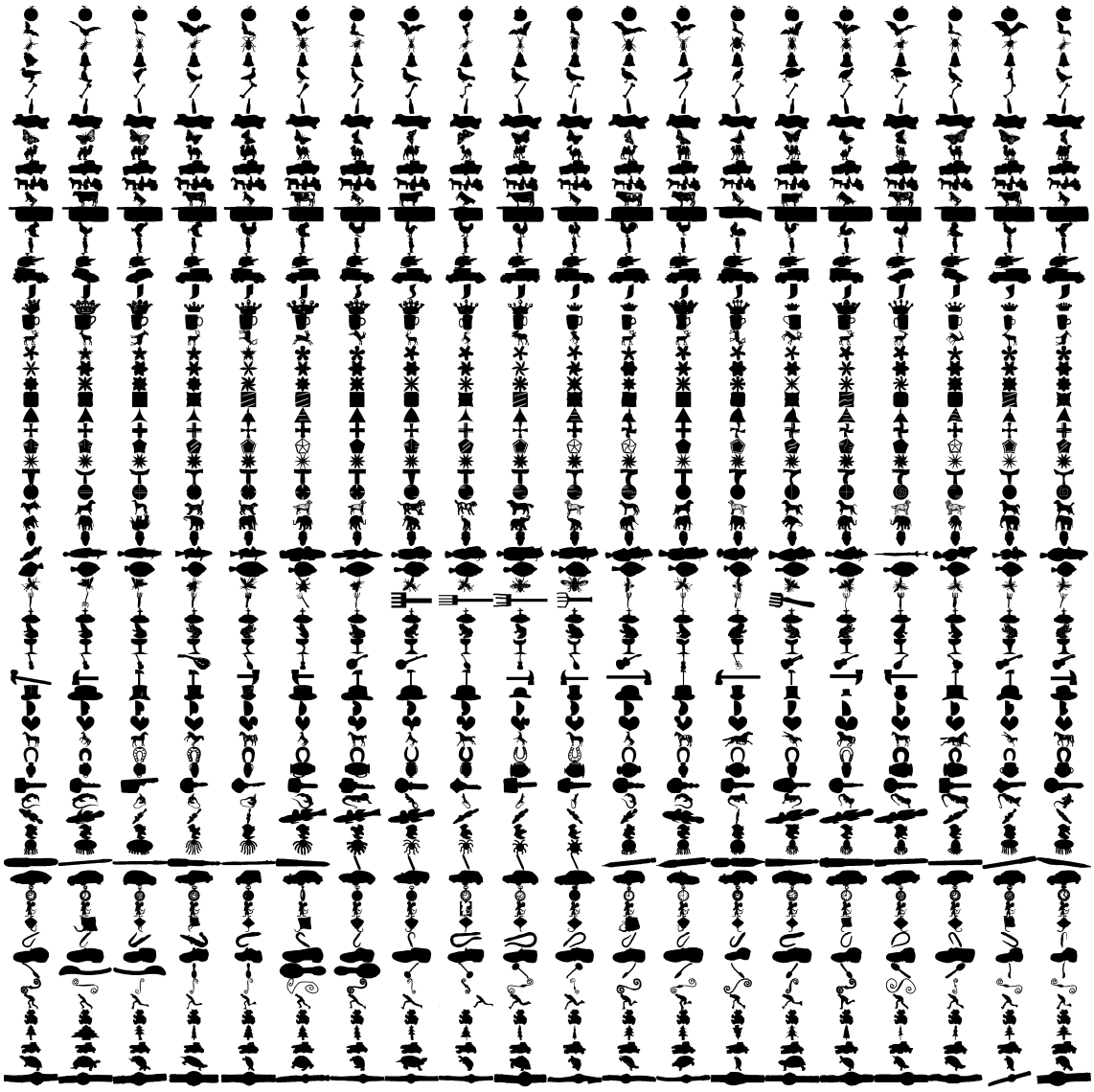


Figure 4.3: MPEG-7 silhouette dataset [117] consisting of 70 shape categories each with 20 different images with high intra-class variability.

from Latecki et al. [117]. All three datasets are binary masks of various shape categories with multiple images per category, see Figure 4.3 for an overview of the MPEG-7.

In Section 4.4.1 we focus on the task of shape retrieval and analyze the influence of the number of sampled points N, the chord offset Δ and fusion of descriptor features on the performance using the KIMIA-25 and KIMIA-99 datasets. In Section 4.4.2 we

analyze the performance of *IS-Match* for the task of shape clustering in combination with various cluster methods.

The first step of our method is to represent the shapes by a sequence of points sampled from the outline of the region. There are two different variants for point sampling: (a) sampling the same number of points from the closed region or (b) equidistant sampling, i. e. fixing the distance between sampled points. The type of sampling significantly influences the invariance properties of our method. Based on equidistant sampling occlusions are ideally handled, but then only shapes at the same scale are correctly matched. By sampling the same number of points our method becomes invariant to similarity transformations, but strong occlusions can disturb the performance. In this chapter we focus on the equidistant sampling for the task of shape retrieval on single scale datasets. Nevertheless all subsequent parts of the method are defined in a manner independent of the sampling type. Therefore, we can switch the sampling type without requiring any modifications of the method.

4.4.1 Shape Region Retrieval

The task of shape retrieval is defined as given a reference shape, rank all other query shapes in order or similarity. We apply our method to shape retrieval on three frequently used datasets. We use the KIMIA-25 and KIMIA-99 from Sharvit et al. [180] and evaluate the influence of parameters, such as the number of sampled points N and Δ for the region shape description. Further, we evaluate on the MPEG-7 silhouette dataset from Latecki et al. [117] and compare the retrieval performance.

4.4.1.1 KIMIA-25 dataset

The first dataset KIMIA-25 consists of 25 images of six different classes. Each shape of the dataset was matched against every other shape of the dataset and the global optimal Salukwadze distance as described in Section 4.3.3 was calculated for every comparison. Then for every reference image all the other shapes were ranked by increasing similarity value. To evaluate the retrieval performance the number of correct first-, second- and third ranked matches that belong to the right class was counted. In all the experiments Δ was set to 5, but experimental evaluations with different parameterizations revealed that changing Δ only has a small effect on shape retrieval performance.

Figure 4.4 illustrates the performance of our algorithm on this dataset, where the sum over all correct first-, second- and third ranked matches is shown. Therefore, the

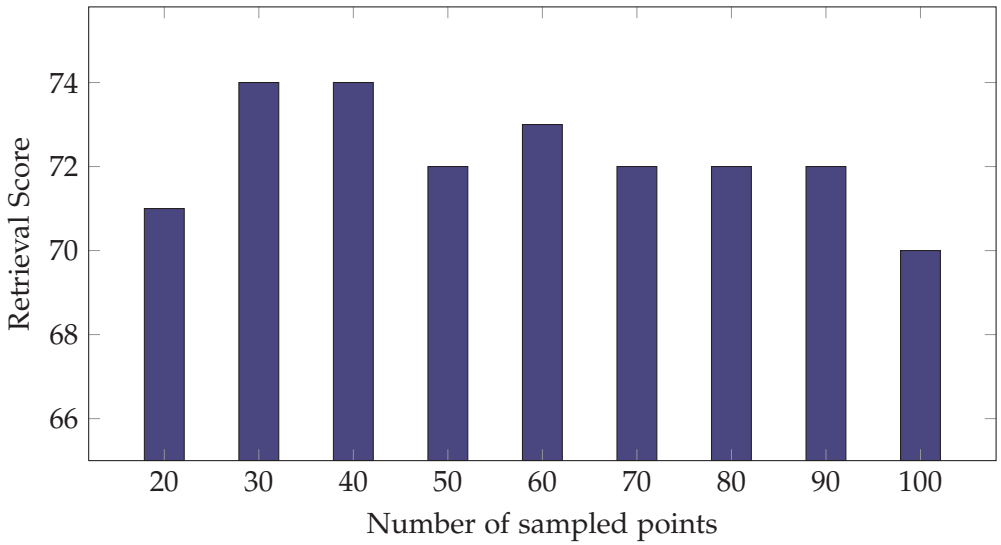


Figure 4.4: Shape Retrieval results in dependence of number of sampled points on the KIMIA-25 dataset [180] consisting of 25 shapes of 6 different classes. Maximum achievable score is 75.

Algorithm	Top 1	Top 2	Top 3	Sum
Sharvit et al. [180]	23/25	21/25	20/25	64
Gdalyahu et al. [82]	25/25	21/25	19/25	65
Belongie et al. [19]	25/25	24/25	22/25	71
Scott and Nowak [175]	25/25	24/25	23/25	72
Biswas et al. [24]	25/25	25/25	23/25	73
Ling and Jacobs [126]	25/25	24/25	25/25	74
Our work	**25/25**	**25/25**	**24/25**	**74**

Table 4.2: Comparison of retrieval rates on the KIMIA-25 dataset of [180]. The number of correct first-, second- and third ranked matches is shown.

best achievable performance value is 75. We present results of *IS-Match* in dependence of the number of sampled points in a range from 20 to 100 sampled points. As can be seen by sampling 30 points, we achieve the highest score of 25/25, 25/25, 24/25 which represents state of the art for this dataset as shown in Table 4.2.

4.4.1.2 KIMIA-99 dataset

The second dataset we analyze is the KIMIA-99 [180], which contains nine classes each consisting of eleven images each. We again performed an all vs. all comparison and ranked the similarity values for each reference shape. The performance of the rank-

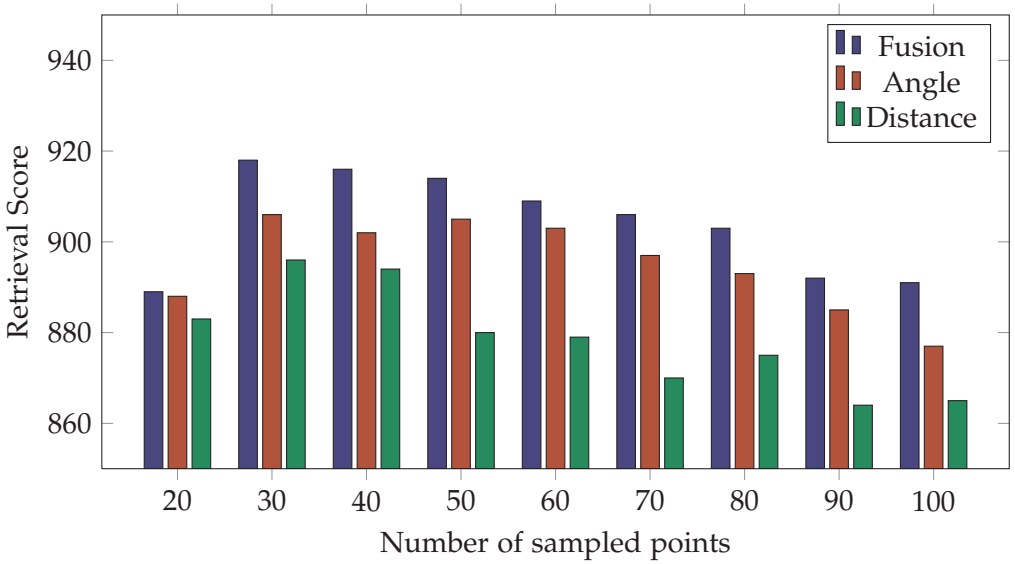


Figure 4.5: Shape retrieval results for different variants on the KIMIA-99 dataset [180] consisting of 99 shapes of nine different classes. Maximum achievable score is 990.

ing properties of our approach is analyzed by evaluating how many of the ten closest matches to each reference image are in the same category.

For this dataset we also analyze the combination of complementary features. For this we calculate the region descriptor using angles as well as distances. Figure 4.5 shows the result for feature combination. Looking at feature alone, the angular features outperform the distance features by a few percent on average. However, when fusing the results the total performance increases. The best fusion result was obtained with an early fusion of the features with equal weighting between them.

Figure 4.5 also shows the effect of scale, when a different number of points are sampled on the region boundary. Again the best performance requires only 30 points and in Table 4.3 we compare our retrieval results to related work for shape retrieval. We again achieve competitive results, e. g. the top-4 ranks in our results were always correct.

In summary, our algorithm achieves state-of-the-art results on the two datasets with only 30 sampled points, which allows to perform a single match within $3\,ms$, outperforming the running time of all other shape matching algorithms by two orders of magnitude. But please note, that since our method heavily depends on the ordering of the sampled silhouette points, it might be quite sensitive to noise. This is not a problem for the clean shapes of the standard shape matching datasets but might be an issue for application in different real-world scenarios.

Algorithm	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
Belongie et al. [19]	97	91	88	85	84	77	75	66	56	37
Kuijper and Olson [111]	99	96	93	93	87	82	75	71	57	56
Tu and Yuille [198]	99	97	99	98	96	96	94	83	75	48
Biswas et al. [24]	99	97	98	96	97	97	96	91	83	75
Ling and Jacobs [126]	99	99	99	98	98	97	97	98	94	79
Sebatian et al. [176]	99	99	99	98	98	98	96	95	94	86
Felzenszwalb et al. [61]	99	99	99	99	99	99	99	97	93	86
Our method	**99**	**99**	**99**	**99**	**98**	**92**	**90**	**85**	**82**	**75**

Table 4.3: Comparison of retrieval rates on KIMIA-99 dataset [180] consisting of 99 shapes of nine different classes.

4.4.1.3 MPEG-7 dataset

As a third dataset we evaluate *IS-Match* on the MPEG-7 silhouette dataset [98, 117] which is one of the most popular dataset for shape matching evaluation. The dataset consists of 70 shape categories, where each category is represented by 20 different images with high intra-class variability. The parametrization of our algorithm is based on the results shown in the previous section.

The overall shape matching performance was evaluated by calculating the so-called bullseye rating, in which each image is used as reference and compared to all of the other images. The mean percentage of correct images in the top 40 matches (the 40 images with the highest shape similarity scores) is taken as bullseye rating.

The measured bullseye rating for *IS-Match* was 84.79% and is compared to state-of-the-art algorithms in Table 4.4. As can be the seen the score is close to the best ever achieved by Felzenszwalb et al. [61] of 87.70%. But please note that [61] uses a more complex descriptor and requires about 500ms per match. Therefore, analyzing the entire dataset takes approximately 136 hours for [61], while with *IS-Match* all similarity scores are provided within a single hour!

Figure 4.6 further illustrates the bullseye scores for all the 70 different classes independently, revealing that most of the classes are retrieved well. Only two classes (Device 6 and Device 9) have a retrieval score below 50%. These are shape categories of the MPEG-7 dataset, where the outer boundary of the shape is different for each image. It makes one wonder what shape really is.

Algorithm	Mokhtarian [143]	Belongie [19]	Scott [175]	Ling [126]	Felzenszwalb [61]	**Our work**
Score	75.44%	76.51%	82.46%	86.56%	87.70%	**84.79%**
Runtime	-	54 h	122 h	84 h	136 h	1 h

Table 4.4: Comparison of retrieval rates and estimated overall runtime in hours for calculating the full $N \times N$ similarity matrix on MPEG-7 dataset consisting of 1400 images showing 70 different classes.

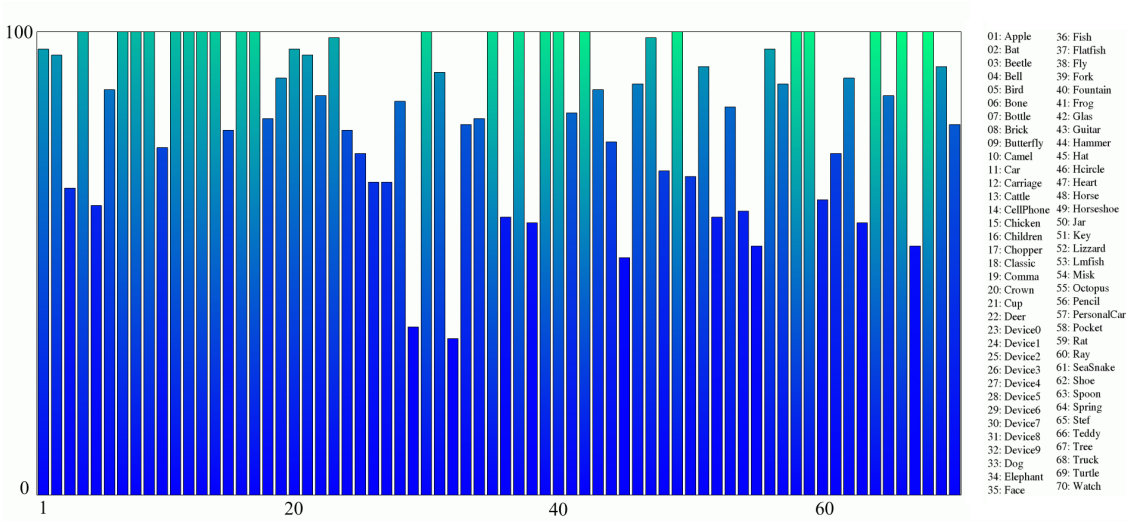


Figure 4.6: Shape matching result for the MPEG-7 silhouette dataset. For each of the 70 different classes the bullseye score, i. e. the number of correct matches in the 40 first ranked shapes, is shown. A total bullseye score of 84.79% is achieved.

4.4.2 Shape Region Clustering

The previous section showed experiments on the shape retrieval datasets, which revealed that the global optimal Salukwadze distance provided by *IS-Match* captures a meaningful notion of shape dissimilarity.

Therefore, we now investigate it for clustering shapes, which has versatile applications in computer vision. For example, shape clusters are used to improve the efficiency of current object detection methods [81, 125] by providing a hierarchical shape structure, which allows to perform detection in a coarse-to-fine approach. It enables automatic labeling in image datasets by outlining existing groups and relations between them [213]. Finding similar shapes also facilitates the unsupervised clustering of object categories.

F-Value / MI	K-Center	Agglom. Clustering	Affinity Propagation
Shape Context	0.52/0.66	0.56/0.70	0.77/0.76
COPAP	0.65/0.76	0.72/0.84	1.00/0.97
Our work	**0.70/0.84**	**1.00/1.00**	**1.00/1.00**

Table 4.5: Comparison of combinations of shape matching and clustering methods on the KIMIA-25 [180] dataset consisting of 25 shapes for 6 classes. The F-values and the mutual information scores are shown.

Shape clustering methods were recently presented by Schmidt et al. [173] who clustered 40 shapes of four classes using dynamic time warping matching and k-means clustering. Yankov and Keogh [213] clustered shapes for grouping together objects in large collections by a manifold clustering approach. We use a two-step approach for shape clustering. First, we build a pairwise similarity matrix by comparing every possible combination of shapes in the input dataset. Second, we apply a pairwise or proximity-based clustering method on the similarity matrix to find the clusters.

The shape similarity measure is not necessarily metric (asymmetric, violation of the triangle inequality) which has to be taken into account by the clustering method. While most classical methods for pairwise clustering [93, 158, 181] only consider symmetric similarity matrices, recent methods as e. g. affinity propagation clustering [78] also work in non-metric spaces. To evaluate the quality of the provided similarity scores of *IS-Match* and to find the best suited clustering algorithm we evaluated all combinations between three shape matching algorithms (Shape Context [19], COPAP [175] and *IS-Match*) and three clustering algorithms (k-center clustering, hierarchical agglomerative clustering and affinity propagation clustering).

K-center clustering starts with a random initialization of the class centers and iteratively refines these centers by decreasing the sum of squared errors. Since it is sensitive to the random initialization it is usually rerun many times to find a good solution.

Hierarchical agglomerative clustering is initialized by setting each data point as an individual class. Then, again it iteratively agglomerates the closest pair of clusters by satisfying a similarity criteria.

Affinity propagation enables clustering of data points analyzing a pairwise similarity matrix. It is based on iteratively exchanging messages between the data points until a good solution emerges. While most clustering methods only keep track of some candidate exemplars during search, affinity propagation considers all data points as candidates and is able to handle non-metric similarity measures. Therefore, its perfectly

F-Value / MI	K-Center	Agglom. Clustering	Affinity Propagation
Shape Context	0.56/0.67	0.65/0.83	0.89/0.87
COPAP	0.56/0.70	0.38/0.61	0.92/0.87
Our work	0.69/0.84	0.81/0.92	**0.97**/**0.96**

Table 4.6: Comparison of combinations of shape matching and clustering methods on the KIMIA-99 [180] dataset consisting of 99 shapes for 9 classes. The F-values and the mutual information scores are shown.

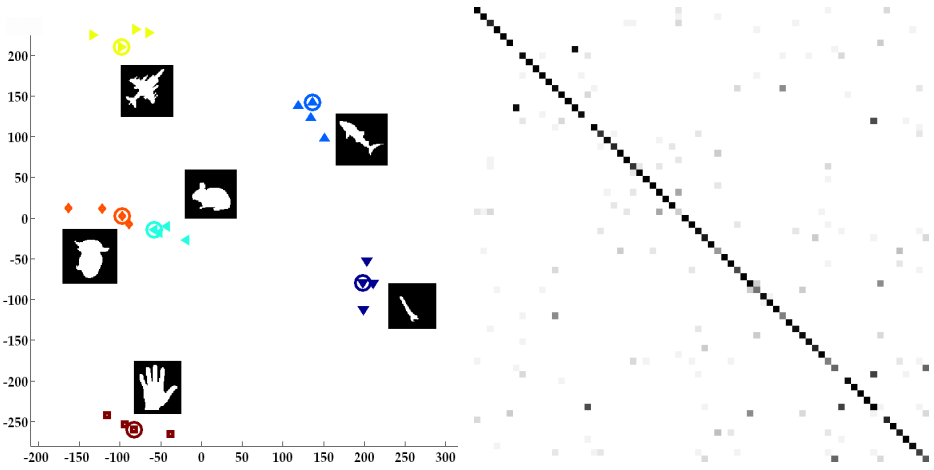
suited for shape clustering, because in general shape similarities do not lie in a metric space, and thus cannot be compared with Euclidian distances.

The performance is analyzed on two datasets, namely the KIMIA-25 and the KIMIA-99, for which the corresponding clustering results for a range of possible combinations of shape matchers and clustering algorithms are shown in Table 4.5 and Table 4.6 respectively. We analyze the results in six combinations of clustering methods and shape matchers, for which we use the default parametrization. Furthermore, for both k-center clustering and hierarchical agglomerative clustering, we set the number of clusters to the true value, whereas for affinity propagation all preference parameters are set to the median of the similarity values, i. e. affinity propagation finds the number of clusters by itself. The k-center algorithm was repeated 10 000 times to cope with the random initialization.

Clustering quality is measured by the F-Value analyzing precision and recall and the information theory based mutual information (MI) value. Therefore, in both cases the higher the value the better the corresponding clustering result. As can be seen *IS-Match* strongly outperforms the Shape Context and COPAP method in terms of clustering quality. Furthermore, affinity propagation clustering leads to better results compared to k-center and hierarchical agglomerative clustering and additionally does not require to fix the number of clusters.

Finally, to demonstrate the high quality and efficiency of the shape clustering method, we applied it to the MPEG-7 dataset. To be able to cluster the entire MPEG-7 dataset (1400 shapes) the full similarity matrix, containing matching results for 1 960 000 shape comparisons (non-symmetric scores), has to be calculated. This is possible using *IS-Match* in only 50 minutes on a single-core desktop PC. Performing clustering on such datasets based on Shape Context or COPAP is not possible in reasonable time.

Figure 4.7(b) shows the confusion matrix of the shape clustering result on MPEG-7 demonstrating excellent clustering quality. The corresponding F-Value is 0.81 and the



(a) KIMIA-25 (b) MPEG-7

Figure 4.7: Clustering results: a) Two-dimensional multi-dimensional scaling visualization of perfect clustering results on KIMIA-25 dataset [180]. Different colors reflect the different clusters identified by affinity propagation. b) Confusion matrix for clustering on MPEG-7 dataset consisting of 1400 shapes for 70 classes. Please note, the results are achieved by affinity propagation without specifying the number of actual clusters.

mutual information value is 0.90. Please note, that these results are achieved without pre-specifying the number of clusters, since affinity propagation finds a reasonable number (91) using default parameters in an autonomous manner.

In order to be able to visualize the results of shape clustering we projected the shapes to a two-dimensional space by applying non-metric multi-dimensional scaling (MDS) to the calculated shape similarity matrix. Because MDS requires a symmetric similarity matrix, we use the minimum distance of the shape pair comparisons. Figure 4.7(a) illustrates the clustering result of *IS-Match* and affinity propagation with color coded cluster assignment, where we get a perfect clustering result (without specifying the number of clusters). Furthermore, as can be seen in Figure 4.8 high quality clustering results are achieved, although MDS is not able to appropriately visualize class assignments.

4.5 Conclusion

In this chapter we introduced a novel method for partial matching of structured descriptors. In contrast to related work which focuses on local interest points and graphical models for structure, we show how the *Structural Measurement Descriptor* (SMD) is

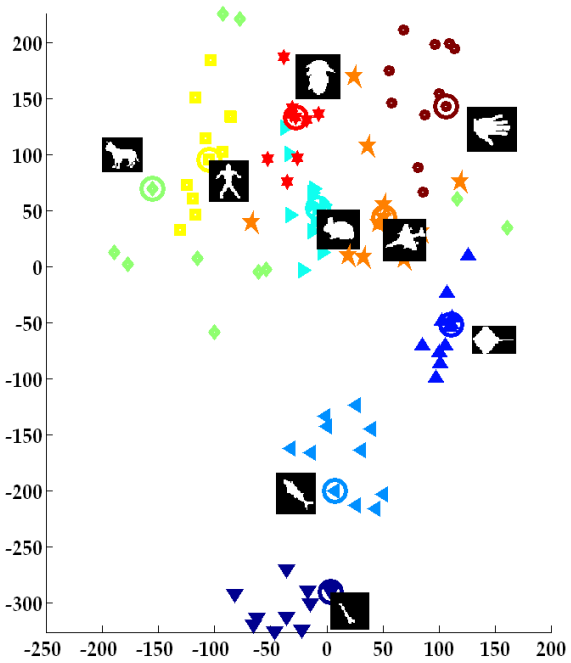


Figure 4.8: Two-dimensional visualization of clustering results on the KIMIA-99 dataset [180] reduced by multi-dimensional scaling (MDS). Different colors reflect the different clusters identified by affinity propagation. The cluster prototype is highlighted by the circles. There are only two false assignments. The other 97 shapes are correctly classified although MDS is not able to appropriately visualize class assignments.

well-suited for enforcing coherent structure. The hierarchical description encodes all parts into one coherent descriptor, where the spatial extent of parts is not known. This is solved by an integral image-based efficient calculation of a 3D tensor containing all combinations for correspondence, length and similarity. We focused on matching closed regions given by a binary mask and proposed a novel method for measuring a balanced partial similarity. The experimental evaluation demonstrates that our partial similarity measure achieves superior computational efficiency of only 3 ms per match as well as competitive results to state of the art in shape retrieval and clustering on the well-known KIMIA-25, KIMIA-99 and MPEG-7 silhouette datasets. Due to the efficiency of the proposed algorithm it is well-suited for real-time applications as e. g. in action recognition by matching human silhouettes to reference prototypes or for tracking applications. Further focus of future work is the task of object detection in cluttered images, which requires a different strategy for extracting and grouping partial shape matches.

When the going gets tough, the tough get going.

Joseph Kennedy

5

Partial Contour Matching for Object Detection

In this chapter we propose a method for object category localization by partially matching image edge contours to a single hand-drawn shape prototype of the category. Previous work in this field either relies on piecewise contour approximations, requires meaningful supervised decomposition, or matches coarse shape-based descriptions at local interest points, as reviewed in Section 5.2. Our method avoids error-prone preprocessing steps by using all obtained edges in a partial contour matching setting. The matched contours are efficiently summarized to long salient matches and aggregated by a star-model to form location hypotheses, as described in Section 5.3. The efficiency and accuracy of our edge contour based voting step yields high quality hypotheses in low computation time. The experimental evaluation achieves excellent performance in the hypotheses voting stage and yields competitive results on challenging datasets like ETHZ Shape classes and INRIA Horses, as shown in Section 5.4.

Contents

5.1 Introduction

Object detection is a challenging problem in computer vision. It allows localization of previously unseen category instances in images. The most common approach involves detecting interest points, which are described over a limited local support window. These descriptions are then compared to a category model, which evaluates the likelihood of an object to be present in the image at various locations. In general, two main paradigms for describing the local interest points are distinguished. First, appearance-based approaches form the dominant paradigm using, for example, the a) orderless bag-of-words model [187], or b) the structured sliding windows [59] and generalized Hough transforms [13] for localization. The former approach analyzes an orderless distribution of local image features and achieves impressive results mainly because of rich local appearance description [141]. The latter approaches also use rich image features, however employ rigid or deformable graphical models to infer the constellation of selected parts. Second, the shape-based detection paradigm has a long history in computer vision and has recently become popular again. Both paradigms have several benefits and drawbacks. For example, shape provides a powerful and often more generic feature [21] since an object contour is invariant to extreme lighting conditions and large variations in texture or color.

In this chapter, we propose a novel method for partial matching of object contours, which is a problem not often investigated so far. Most related work can be described in four categories of breaking down full contour. The research [153, 182] focuses on the aspect of learning local sets of contours in codebooks, where chamfer matching is used to evaluate local shape similarity. Other research uses piecewise approximations of contour by short segments [65, 162] or employ manual supervision to derive decompositions [218, 220]. In [20, 134, 152] the problem is cast as a graph matching between

shape-based descriptors on local interest points. Hence, most approaches deal with local parts already and incorporate graphical models for grouping the local features into a coherent model.

The main motivation for our method is the Gestalt principle of continuation [104], which states that the human eye is compelled to follow lines and curves. This flow along a line enables us to complete missing information. Such *"connectedness is a fundamental powerful driving force underexploited in object detection"* [63]. Viewing edge contours as connected sequences of any length instead of short segment approximations or local patches on interest points provides more discrimination against background clutter.

In this chapter we focus on the partial matching of noisy image edges to relax the constraints on local neighborhoods and to assign entire edges as background disregarding local similarities. We propose a category localization method which efficiently retrieves partial edge contours which are similar to a single reference prototype. We introduce a self-containing hierarchical description for edge contours. This enables partial matching and an efficient selection and aggregation of partial matches to identify and merge similar overlapping contours up to arbitrary length. A key benefit is that the longer the matches are, the more they are able to discriminate between background clutter and the object instance. In this way we lift standard figure / ground assignment to another level by providing local similarities for all edges in an image. We retrieve these partial contours and combine them directly in a similarity tensor.

Many complex graphical models exist to group local features into a coherent category model, for example the pictorial structures [60, 73]. In this chapter we employ an ISM-style clustering-based center voting step [120] to hypothesize object locations. Due to our long salient partial matches, this greatly reduces the search space to a handful of hypotheses and shows excellent performance compared to state of the art in the voting stage. For a full system evaluation, the hypotheses are further verified by a standard multi-scale histogram of gradients (HOG) classifier.

5.2 Related Work

There exists a long history of work in the shape-based paradigm which achieved state-of-the-art performance over the years for several object categories using contour information. See Figure 5.1 for a brief overview of the most related approaches of matching contours in images. The research falls into four main categories, namely (i) learning codebooks of contour fragments, (ii) approximating contours by piecewise segments,

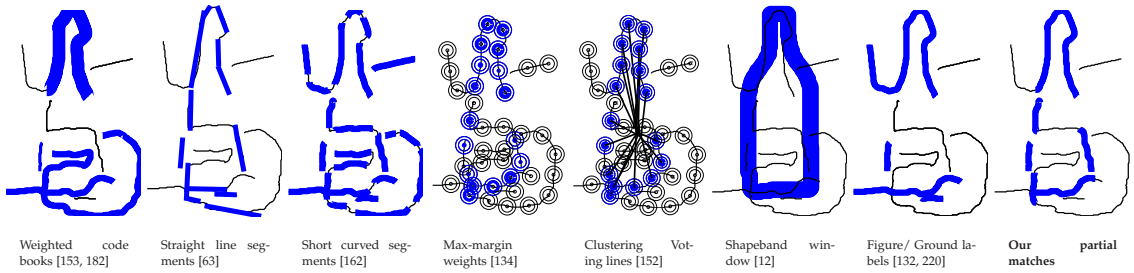


Figure 5.1: Overview of related work: Our approach relaxes the piecewise approximations and local neighborhoods. We use partial matching to find contour fragments belonging to the foreground rather than discarding entire edges, see Section 5.2.

(iii) using local description of the contour at selected interest points, or (iv) assigning entire edges to either foreground or background. Additional techniques are used, for example learning deformation models, sophisticated cost functions or probabilistic grouping. In this chapter, we focus the review of related work on the aspect of the contour description and matching. In the next chapter, we discuss the aspect of discriminative learning of category models.

5.2.1 Learning codebooks

Shotton et al. [182] and Opelt et al. [153] concurrently proposed to construct shape fragments tailored to specific object classes. Both find matches to a pre-defined fragment codebook by chamfer matching to the query image and then find detections by a star-shaped voting model. Their methods rely on chamfer matching which is sensitive to clutter and rotation. In both approaches the major aspect is to learn discriminative combinations of local contour parts as weak classifiers using boosting to build a strong category model detector.

5.2.2 Piecewise approximation

Ferrari et al. [63, 66] build groups of approximately straight adjacent segments (kAS) to work together in a team to match the model parts. The segments are matched within a contour segmentation network which provides the combinations of multiple simple segments using the power of connectedness. In later work they show how to automatically learn segment codebooks [63], and how to learn category shape models from training images [64]. In a verification step they use a thin-plate-spline (TPS) matching to accu-

rately localize the object boundary. Similar to this, Ravishankar et al. [162] use short segments to approximate the outer contour of objects. In contrast to straight segments, they prefer slightly curved segments to have better discriminative power between the segments. They further use a sophisticated scoring function which takes local deformations in scale and orientations into account. However, they break the reference template at high curvature points to be able to match parts, again resulting in disjoint approximations of the actual contour. In their verification stage, the gradient maps are used as underlying basis for object detection avoiding the error-prone detection of edges.

5.2.3 Shape-based interest points

This category uses descriptors for interest points to capture and match coarse descriptions of the local extent of shape around sparse landmark points in the image.

Leordeanu et al. [122] use geometric features based on normal orientations and pairwise interactions between them to learn and detect object models in images. Their simple features are represented in pairwise relations in category specific models that can learn hundreds of parts. Berg et al. [20] formulate the object detection problem as a deformable shape matching problem. However, they require hand-segmented training images and do not learn deformation models in training. Fidler et al. [67, 72] show a method for hierarchical statistical learning of local configurations of Gabor filters for category classification, which is later extended to generic *shapinals* as layer-independent shape terminals [68], and further for object detection by generic shape vocabulary learning [70, 71].

Further in the line is the research of Maji and Malik [134] and Ommer and Malik [152] which match richer Geometric Blur features to training images. The former use a max-margin framework to learn discriminative weights for each feature type to ensure maximal discrimination during the voting stage. The latter provide an interesting adaptation of the usual Hough-style center voting. Ommer and Malik transform the discrete scale voting to a continuous domain where the scale is another unknown in the voting space. Instead of multiple discrete center vectors, they formulate the votes as lines and cluster these to find scale-coherent hypotheses. The verification is done using a HOG-based fast Support Vector Machine kernel (IK-SVM).

5.2.4 Figure / ground assignment

Similar in concept but not in practice is the research of Zhu et al. [220] and Lu et al. [132]. They cast the problem as figure / ground labeling of edges and decide for a small set of long edges which of these belong to the foreground and which are background clutter. By this labeling they reduce the clutter and focus on salient edges in their verification step. Lu et al. use particle filters under static observation to simultaneously group and label the edge contours. They use a shape descriptor based on angles to decide edge contour similarity as a whole. Zhu et al. use landmark points along the reference contour to find possible edge contour combinations and then solve a cost function efficiently using linear programming. They find a maximal matching between a set of query image contours and a set of salient contour parts from the reference template, which was manually split into a set of finite length reference segments.

Both assume to match entire edge contours to the reference sets and require long salient contours. Recent work by Bai et al. [12] is also based on a background clutter removal stage called shapeband. Shapeband is a new type of sliding window adapted to the shape of objects. It is rigid yet includes separation of foreground and background to provide location hypotheses and to select edge contour candidates. However, in their runtime intensive verification step they iteratively compute shape context descriptors [19] to select similar edge contours as a whole. Another recent approach by Gu et al. [90] proposes to use regions instead of local interest points or contours to better estimate the location and scale of objects.

5.3 Partial Contour Matching for Object Detection

In this chapter we propose a novel method for partial matching of noisy contour edges to relax the constraints on the neighborhood. Contrary to local interest points and figure / ground assignment of entire edges, we do not have a finite length descriptor of an edge contour. This is a major challenge of partial matching as one needs to measure local and global similarities of the contours and then extract partial matches. In contrast to region-based similarity measures, one cannot aim for a balance between contour similarity and partiality, as an arbitrary part of the contour may be background noise. We use edge contours in the query image and match them at any length from short contour segments up to full regions boundaries using partial shape matching. In such

a setting the similarity to the prototype shape decides the complexity and length of the considered contours.

In the following we describe our approach to detect objects by computing partial similarities between edge contours in a query image and a reference template. We define the following terminology to describe each aspect of the detection steps. We use the term *query contour* to refer to any extracted linked edge from an image, while we use a *partial contour* to denote a part of such an edge contour. *Contours* have quite an arbitrary nature and require special care due to its sources of noise from low-level linking or occlusions. First, contours can be arbitrarily long as the perceptual grouping properties of edge linking process determines the starting and end points of a contour. Second, contours may contain irrelevant parts, may be incomplete due to missing edge detection responses, or may contain occlusions which make parts of the object invisible. In this chapter, we use a single reference contour for comparison, which is a single hand-drawn model of the object's outer boundary. A valid match is defined as a correspondence of parts, which are similar between a partial contour of the query and that of the reference contour.

Our goal is to identify matches from the query contours of arbitrary length (contained within the query edges) to the reference contour, by analyzing a self-contained representation and description of the shape of the detected edge contours. We exploit the representation containing the whole as well as any part of a contour, which enables matching independently from the remaining parts.

Our detection method consists of four parts. First, edges are extracted from an image and linked as lists of coordinates. Second, this representation is described using the structural measurement descriptor for contours. Third, the partial similarity between the reference contour and all query contours are evaluated and valid matches are grouped using an analysis of the tensor space. Fourth, each valid match has a correspondence to the reference contour, which is used to infer a center vote to estimate the location of the searched object and sixth, aggregate coherent fragments based on their voting, scale and correspondence to the reference contour.

5.3.1 Edge Extraction

For the task of category object detection, we are dealing with noisy cluttered image. The images contain one or multiple objects of the category in search, but for the most part the image contains background clutter. Contrary to the classical interest point detection

methods, our method is designed for binary images without additional edge weights, which can however be included later on, and uses all image edges densely. For this we extract binary edges from the images by using an edge extraction and linking methods. The result is a set of query image edge lists, which define the edges as a sequence of ordered points. Specifically, we employ apply the Berkeley edge detector [137] and link the results to a set of coordinate lists [109], through any other method may also be employed. For the obtained query contours, points are sampled at equal distance, resulting in a sequence of points $C = (P_1, P_2, \ldots, P_M)$ per contour, where the contour is open, i. e. no points are used twice from the contour points.

5.3.2 Contour Description

Since we are analyzing up to hundreds of query contours for partial matches, we require an efficient description and matching. Only this will allow to compare every edge contour in an image to the given reference contour in reasonable time. The structural measurements descriptors and tensor similarity distance is the powerful and low complexity method required to perform partial contour matching of this magnitude as it only requires about 3 ms per match (as opposed to the fastest related work of 200 ms).

The sampling distance d between the points allows to handle different scales. Sampling with a larger distance equals to a larger scale factor, and vice versa. For detecting objects in query images we perform an exhaustive search over a range of scales, which is efficiently possible due to the properties of our descriptor and matching method.

As a first step, we sample points are at equal distance d_r from the reference contour, which is denoted as $\mathcal{R} = (P_1, P_2, \ldots, P_N)$. For the obtained query contours, also equidistant points are sampled, resulting in a sequence of points $\mathcal{C} = (P_1, P_2, \ldots, P_M)$ per contour, where N and M are the length of the contours. Changing the sampling distance d_c between the points of the query contour allows to handle different scales. Sampling with a larger distance equals to a larger scale factor, and vice versa. For detecting objects in query images we perform an exhaustive search over a range of scales, which is efficiently possible due to the properties of our descriptor and matching method. The details for selecting the sampling distance d_c are described in the experimental section.

We use a the Structural Measurement Descriptor for contours, which encode the geometry of the sampled points leading to a translation and rotation invariant description for a query contour. The descriptor is calculated from the relative spatial orientations between lines connecting the sampled points. We calculate only angles α_{ij} between a

line connecting the points P_i and P_j and a line to a third point relative to the position of the previous two points. The measurement for an angle α_{ij} is calculated between the two chords $\overline{P_iP_k}$ and $\overline{P_jP_k}$ and P_k is again Δ positions away as defined by

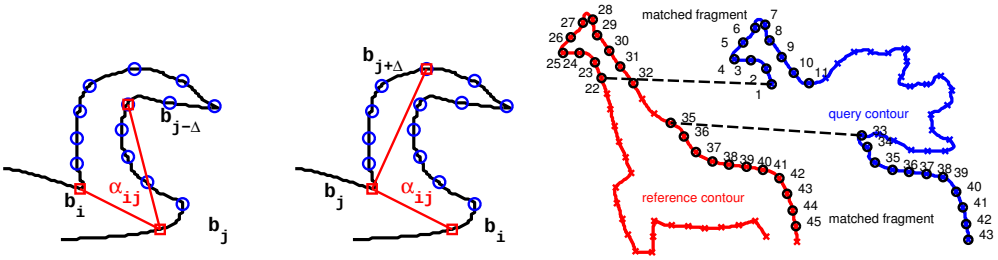
$$\alpha_{ij} = \begin{cases} \sphericalangle(\overline{P_iP_j}, \overline{P_iP_{j-\Delta}}) & \text{if } i < j \\ \sphericalangle(\overline{P_iP_j}, \overline{P_iP_{j+\Delta}}) & \text{if } i > j \\ 0 & \text{if } abs(i-j) \leq \Delta \end{cases}, \tag{5.1}$$

where P_i and P_j are the i^{th} and j^{th} points in the sequence of sampled points of the contour and Δ is an offset parameter of the descriptor. The value of Δ is fixed to 5 for all experiments. See Figure 5.2 for an illustration of the choice of points along the contour. The third point is chosen depending on the position of the other two points to ensure that the selected point is always inside the contour. This allows us to formulate the descriptor as a self-containing descriptor of any of its parts. It is important for contours as the source of information for the boundary stems from from underlying image edges. Hence the boundary of the shape is often broken up into parts or contaminated with noise and occlusion contours.

The angles α_{ij} are calculated between every pair of points along a contour. In such a way a contour defined by a sequence of M points is described by an $M \times M$ matrix where an entry in row i and column j yields the angle α_{ij}. The proposed descriptor has four important properties. First, its angular description makes it translation and rotation invariant. Second, a shift along the diagonal of the descriptor handles the uncertainty of the starting point in edge detection. Third, it represents the connectedness of contours by using the sequence information providing a local (close to matrix diagonal) and global (far from matrix diagonal) description. And most importantly, the definition as a self-containing descriptor allows to implicitly retrieve partial matches, which is a key requirement for cluttered and broken edge results.

5.3.3 Valid Contour Matches

Matching and merging partial contours is an important part of our approach and is based on the Structural Measurement Descriptor for contour defined in the previous section. For any two descriptors representing two contours, the aim of matching is to identify parts of the two contours which are similar to each other. In terms of comparing corresponding descriptor matrices, one has to compare all sub-blocks of the descriptor



(a) $i < j$ (b) $i > j$ (c) Partial matches

Figure 5.2: Illustration of 2D angle description and matching. a-b) An angle is measured between any two sampled points b_i and b_j, which define a fragment inside an edge contours, c) shows partial matches in an occluded edge to a reference contour.

matrices to find all matching possibilities and lengths. For efficient calculation of all similarity scores, we apply the tensor similarity method from Chapter 4 to access the partial descriptor differences in constant time, which returns the similarities (differences between our angle descriptors) for all matching triplets $\{r, q, l\}$ stored in a 3D similarity and correspondence tensor $\Theta(r, q, l)$. The first two dimensions identify the starting points of the match in the reference (r) and query contour (q) and the third dimension defines the length (l) of the match. Note that this tensor fully defines all possible correspondences between the reference and the edge contour. However, the length is constrained to the longest possible length given the total length of each contour and the respective starting point. However, since we are dealing with closed regions, matching may overlap across the first point in the sequences of points. Hence, the maximum length is equal to the full length of shapes. Hence, the reduced tensor similarity $\Theta(r, q, l)$ is defined as

$$\Theta(r, q, l) = \begin{cases} \Delta^n & \text{if min(r+l,q+l) < min(M,N)} \\ \infty & \text{otherwise} \end{cases} \tag{5.2}$$

where M and N are the number of points in the open contours for reference and query, which limit the possible correspondences. Figure 5.3 shows the similarity tensor for the partial matching example in Figure 5.2. The two matched fragments correspond to the peaks (a) in the tensor, here a single slice at a fixed length $l = 11$ is shown.

The main challenge lies in the immense redundancy within the tensor as there may be many overlapping and repetitive matches. For a given reference contour of N points and a query contour of M points, there are $\sum_l \sum_q (N - M + 1 + q - l)$ possibilities for

matching arbitrary lengths of the two contours. This poses a problem for object detection in cluttered images and collecting a minimal set of coherent. Our goal is to find the longest and most similar matches and merge repetitive matches instead of retrieving all individual matches. First, it is necessary to outline some of the properties of the 3D similarity and correspondence tensor $\Theta(r, q, l)$. The following are observations about the nature of the 3D tensor:

I. **A partial contour** (r, q, l) is assigned a similarity (Euclidean distance between angular descriptors) by $\Theta(r, q, l)$.

II. **Length variations** (r, q, l_2) with $l_2 < l$ define the same correspondence, yet shorter in length.

III. **Diagonal shifts** in the indices $(r + 1, q + 1, l)$ also represent the same match, yet one starting point *later*.

IV. **Unequal shifts** $(r + 1, q, l)$ define a different correspondence, however very similar and close.

V. **Occlusions or noise** cause multiple matches per edge contour to exist. The example in Figure 5.2 is a shifted match *much later* $(r + 13, q + 32, l)$ defining the same correspondence, yet skipping (32-13=19) points of noise.

VI. **Matches near the end** of each contour (if not closed) have a maximal length given by the remaining points in each contour sequence.

Perfect matches would result in singular *peaks* in a slice. However due to these small shifts along the same correspondence or with an unequal offset, matches result in a *hill*-like appearance of the similarity, see Figure 5.3(a). Given these properties we now define a matching criterion to deliver the longest and most similar matches, i. e. finding the peaks not once per slice but for the entire 3D tensor. This summarization is made of three steps: (a) finding valid correspondences satisfying the constraints on length and similarity, (b) merging all valid correspondences to obtain the longest combination of the included matches (property II) and (c) selecting the maximal similarity of matches in close proximity (property IV). The steps are in detail as following. First, we define a function $\mathcal{L}(r, q, l)$ which gives the lengths at a given valid correspondence tuple (r, q) as

$$\mathcal{L}(r, q, l) = \begin{cases} l & \text{if } \Theta(r, q, l) \leq s_{lim} \text{ and } l \geq l_{lim} \\ 0 & \text{else} \end{cases}, \tag{5.3}$$

where the value at $\mathcal{L}(r,q,l)$ is the length of a valid fragment. A valid fragment has a similarity score below the limit s_{lim} and a minimal length limit of l_{lim}. This function is used to define a subset of longest candidates by

$$\Gamma(r,q) = \forall_{r,q} : \underset{l \in \min(N,M)}{\arg\max}\ \mathcal{L}(r,q,l), \tag{5.4}$$

where $\Gamma(r,q)$ is a subset of $\Theta(r,q,l)$ containing the longest matches at each correspondence tuple (r,q). This set contains matches for every possible correspondence given by the constraints on similarity and matching positions (see property II, VI). However, we further want to reduce this to only the local maxima (conserving property IV). Since the set can now be considered as a 2D function, we find the connected components $\mathcal{C}$ satisfying $\Gamma(r,q) > 0$. The final set of candidates are the maxima per connected component and is defined as

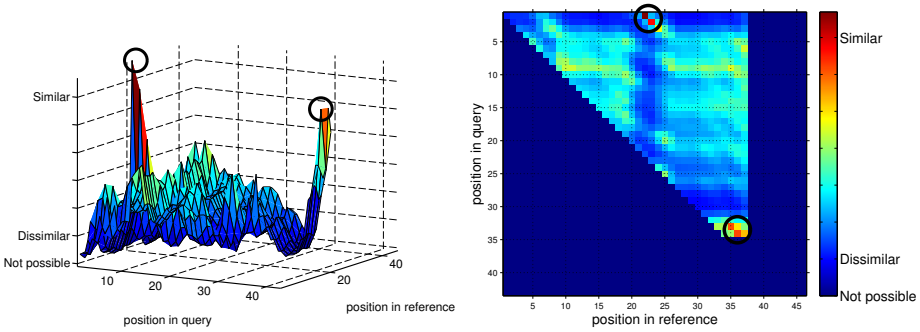
$$\Upsilon(r,q,l) = \forall c_i \in C : \underset{\Theta(r,q,l)}{\arg\max}\,(\Gamma(r,q) \in c_i), \tag{5.5}$$

where $\Upsilon(r,q,l)$ holds the longest possible and most similar matches given the constraints on minimum similarity s_{lim} and minimal length l_{lim}. In the example shown in Figure 5.2 and 5.3 the final set contains two matches, which are the longest possible matches. Note that shorter matches in the head and back are possible, but are directly merged to longer and more discriminative matches by analyzing the whole tensor.

Furthermore, obtained matches are local maxima concerning similarity scores. This provides an elegant and efficient summarization leading to coherent and discriminative matches and reduced runtime.

5.3.4 Hypothesis Voting

Matching as described in the previous section provides a set of matched partial contours for the query edges. Since the object contours in the image are likely to be broken into multiple edge contours, these matches have to be combined as well to form object location hypotheses. In the following we describe how matched contours are grouped for object locations hypotheses and scores are calculated.



(a) Similarity scores in 3D

(b) A single slice in 2D

Figure 5.3: Illustration of the similarity and correspondence tensor $\Theta(r, q, l)$ at length $l = 11$ for example shown in Figure 5.2(c): (a) the two peaks correspond to the matches found. Matching uncertainty results in multiple peaks in a *hill*-like appearance. (b) shows the same similarity in a flat view, where red signals high similarity and dark blue defines invalid matches due to length constraints.

5.3.4.1 Contour Aggregation

Up to this point we have a set of triplets $\Upsilon(r, q, l)$ of matched partial contours of image edges detected in a query image which are highly similar to the provided reference contour. Every match has a certain similarity and length. For grouping together multiple matches we use a star-shape model for center voting. We calculate the center votes for every contour point on the reference contour to the mean of all reference contour points. We then map each matched contour to its reference contour and estimate the object centroid from the given correspondence triplet, by a generalized Hough transformation [13] defined by

$$\mathcal{H} : (r = q, R) \rightarrow R_r - \frac{1}{N} \sum_{i}^{N} \mathcal{R}_i) \tag{5.6}$$

where $\mathcal{H}$ is the center vote for a given correspondence (q, r) from the mean position over all reference contours points $\mathcal{R}$. The aggregation of the individual contour matches identifies groups of contours which compliment each other and form a more complete object location hypothesis. For this step we cluster the matched fragments analyzing their corresponding center votes and their scale by mean-shift mode detection [39] with a scale-dependent bandwidth. The bandwidth resembles an analogy to the classical

Hough transformation [55, 95] accumulator bin size, however with the added effect that we combine the hypotheses locations in the continuous domain rather than discrete bins and the mean-shift automatically detects the number of modes, i. e. location hypotheses.

5.3.4.2 Hypothesis Ranking

All obtained hypotheses are ranked according to a confidence. For this purpose we investigate two ranking methods. The first is based on the coverage of detected fragments, where ζ_{COV} is a score relative to the amount of the reference contour that is covered by the matched fragments, defined as

$$\zeta_{COV} := \frac{1}{N} \sum_{i=1}^{N} (f_i \times S_i), \tag{5.7}$$

where f_i is the number of times the i^{th} contour point has been matched and S_i is the corresponding weight of this point. This is normalized by the number of contour points N in the reference contour. The coverage score ζ_{COV} provides a value describing how many parts are matched to the reference contour for the current hypothesis. In this chapter, we use a uniform weight of $S_i = 1$. However, contour flexibility [209] or discriminative importance of the reference contour can provide these weights.

As a second score, we evaluate a ranking as proposed by Ommer and Malik [152]. They define the ranking score ζ_{PMK} by applying an SVM classifier to the image windows around the location hypotheses. The kernel is the pyramid match kernel (PMK) [89] using histograms of oriented gradients (HOG) as features. Positive samples for each class are taken from the ground truth training set. Negative samples are retrieved by evaluating the hypotheses voting and selecting the false positives. The bounding boxes are resized to a fixed height while keeping median aspect ratio. Since the mean-shift mode detection may not deliver the true object location, we sample locations in a grid, a fourth of width and height to each side, of windows around the mean-shift center. At each location we evaluate the aforementioned classifier and retrieve the highest scoring hypothesis as new detection location.

5.4 Experimental Evaluation

We demonstrate the performance of our proposed object category localization method on two different shape datasets: ETHZ Shape (Section 5.4.1) and INRIA Horses (Sec-

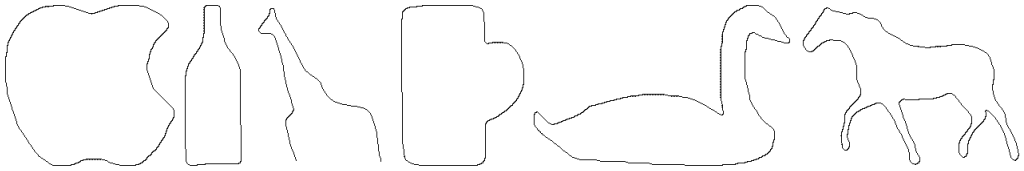


Figure 5.4: Reference contours for the ETHZ Shape and the INRIA Horses datasets.

tion 5.4.2). For each dataset we use a single reference prototype as category more, which is used to partially match again the image contours. See Figure 5.4 for an overview of the reference prototypes used. The first five are the provided hand-drawn models in the ETHZ Shape classes, while the right-most is an example segmentation of a single horse, see Figure 5.6 for some exemplary successful detections and some failure cases.

We significantly outperform related methods in the hypotheses generation stage, while attaining competitive results for the full system. Results demonstrate that exploiting the connectedness of edge contours in a partial contour matching scenario enables to accurately localize category instances in images in efficient manner. Gradient magnitude information, although unreliable since its originates from image contrast, plays important role in related work [63, 134, 152, 162] and could be integrated in future work.

For all experiments the parameters are as follows: The offset Δ is set to 5 for a fine grained descriptor. The similarity limit is $s_{lim} = 0.005$ and the minimum length is $l_{lim} = 0.3 * N$, where N is the number of points in the reference contour. The contour matching is not inherently scale invariant. We analyze 10 scales per image, where scale is defined by the distance $d = d_N \times s$ between the sampled points and s is a scale from a set of 10 neighboring octaves of $2^{1/4}$. The mean-shift bandwidth $w_b = 10 \times s$. Localization of an object over all scales (!) requires on average only 5.3 seconds per image for ETHZ Shape dataset in a Matlab implementation.

5.4.1 ETHZ Shape Classes

The ETHZ Shape classes is a challenging dataset introduced by [66] for evaluating shape detection methods. It consists of five object classes (applelogos, bottles, giraffes, mugs and swans) and a total of 255 images. All classes contain intra-class variations (especially giraffes and mugs) and significant scale changes. The images sometimes contain multiple instances of a category and have a large amount of background clutter. This dataset has received a lot of research focus, which shows in the large range of competitive work and different test protocols, complexity of category models and mixtures of shape and appearance features.

	Voting and Ranking Stage (FPPI=1.0)						*Verification Stage (FPPI=0.3/0.4)*				
ETHZ Classes	Hough [65]	w_{ac} [152]	M^2HT [134]	**Our work**	PMK [152]	**Our work**	M^2HT [134]	PMK [152]	KAS [63]	System Full [65]	**Our work**
Apples	43.0	80.0	85.0	90.4	80.0	90.4	95.0/95.0	95.0/95.0	50.0/60.0	77.7/83.2	93.3/93.3
Bottles	64.4	92.4	67.0	84.4	89.3	96.4	92.9/96.4	89.3/89.3	92.9/92.9	79.8/81.6	97.0/97.0
Giraffes	52.2	36.2	55.0	50.0	80.9	78.8	89.6/89.6	70.5/75.4	49.0/51.1	39.9/44.5	79.2/81.9
Mugs	45.1	47.5	55.0	32.3	74.2	61.4	93.6/96.7	87.3/90.3	67.8/77.4	75.1/80.0	84.6/86.3
Swans	62.0	58.8	42.5	90.1	68.6	88.6	88.2/88.2	94.1/94.1	47.1/52.4	63.2/70.5	92.6/92.6
Average	53.3	63.0	60.9	**69.4**	78.6	**83.2**	91.9/93.2	87.2/88.8	61.4/66.9	67.2/72.0	**89.3/90.5**

Table 5.1: Hypotheses voting, ranking and verification stages show competitive detection rates using PASCAL criterion for the ETHZ Shape dataset [66] compared to related work. For the voting stage our coverage score increases the performance by 6.5% [152], 8.5% [134] and 16.1% [65] leading to state-of-the-art voting results at reduced runtime.

In general, for the ETHZ Shape classes there exist two main methods for evaluation. First, a class model is learned by training on half of the positive examples from a class, while testing is done on all remaining images (half of positive examples and all other negative classes) averaged over five random splits. Second, the ETHZ dataset additionally provides hand-drawn templates per class to model the categories. This step requires no training and has shown to provide slightly better results in a direct comparison [65]. Further, the detection performance may be evaluated using one of the two measures, namely the stricter 50%-PASCAL or the 20%-IOU criterion, which require that the intersection of the bounding box of the predicted hypotheses and the ground truth over the union of the two bounding boxes is larger than 50% or 20% respectively. Additional aspects in the evaluation are the use of 5-fold cross validation, aspect ratio voting and most influential the use of features. Using strong features including color and appearance information naturally has a benefit over gradient information and again over pure binary shape information, which is the case for our method.

The goal of the partial matching in this chapter is to evaluate how well one can detect objects by partially matching their outlines. Thus in our approach we use the hand-drawn models to match only binary edges in an query image and for a full system including geometric and appearance verification, we verify the location hypotheses using a standard gradient-based classifier trained on half of the positive training samples (PMK) and average over five random runs of training sample selection.

Class-wise results for ETHZ Shape classes using the strict PASCAL criterion are given in Table 5.1. The focus of our method lies on hypothesis voting stage, where we can show excellent results of 69.4% and 83.2%, without and with a PMK classifier

ETHZ Shape classes: *Verification Stage (FPPI = 0.3/0.4) using 20%-IoU*						
Supervised Lu [132]	Template Ravishankar [162]	Template Ferrari [66]	Template Ferrari [64]	Codebook Ferrari [63]	Learned Ferrari [64]	Template+Learned **Our work**
90.3/91.9	93.0/95.2	70.5/81.5	82.4/85.3	74.4/79.7	71.5/76.8	**94.4/95.2**

Table 5.2: Average detection rates for related work on hand-drawn and learned models.

ranking. The PMK ranking increases the scores for three classes (bottles, giraffes and mugs). The reason is that the classifier is better able to predict the instance of these classes, especially for mugs, where our system produces twice as many hypotheses compared to the other classes (on average 20 for mugs compared to 8 for the other classes). The coverage score performs better on compact object classes (applelogos and swans). See Figure 5.5 for the precision/recall curves for the ETHZ Shape classes.

We achieve an overall improvement over related work ranging from 6.5% [152], 8.5% [134] to 16.1% [65] without classifier ranking, and 4.6% over [152] using a classifier ranking. Some of these methods do not use hand-drawn prototypes. We also achieve competitive results after verification of 90.5% compared to 66.9% [63], 72.0% [65], 88.8% [152] and 93.2% [134] at 0.4 FPPI.

Due to the lack of separate hypothesis voting results for other approaches, we also provide a range of comparisons with previous work using the full system. We evaluate our method using the 20%-IOU criterion and summarize the results in Table 5.2. Compared to these related work we also achieve excellent results using this criterion.

5.4.2 INRIA Horses

As a second dataset we use the INRIA Horses [65, 101], which consists of 170 images with one or more horses in side-view at several scales and cluttered background, and 170 images without horses. We use the same training and test split as [65] of 50 positive examples for the training and test on the remaining images (120+170). We again use only a single reference template which was chosen from the pixel-wise segmentation of a random horse from the training set. For this dataset the performance is 83.72% at FPPI=1.0 and thus is better than recent results 73.75% by [64], 80.77% by [63] and almost as good as 85.27% by [134], which additionally vote for aspect ratios. Presumably this would also increase our recall for the strongly articulated horses, since a single rigid reference contour does not capture the articulation and hence center vote changes.

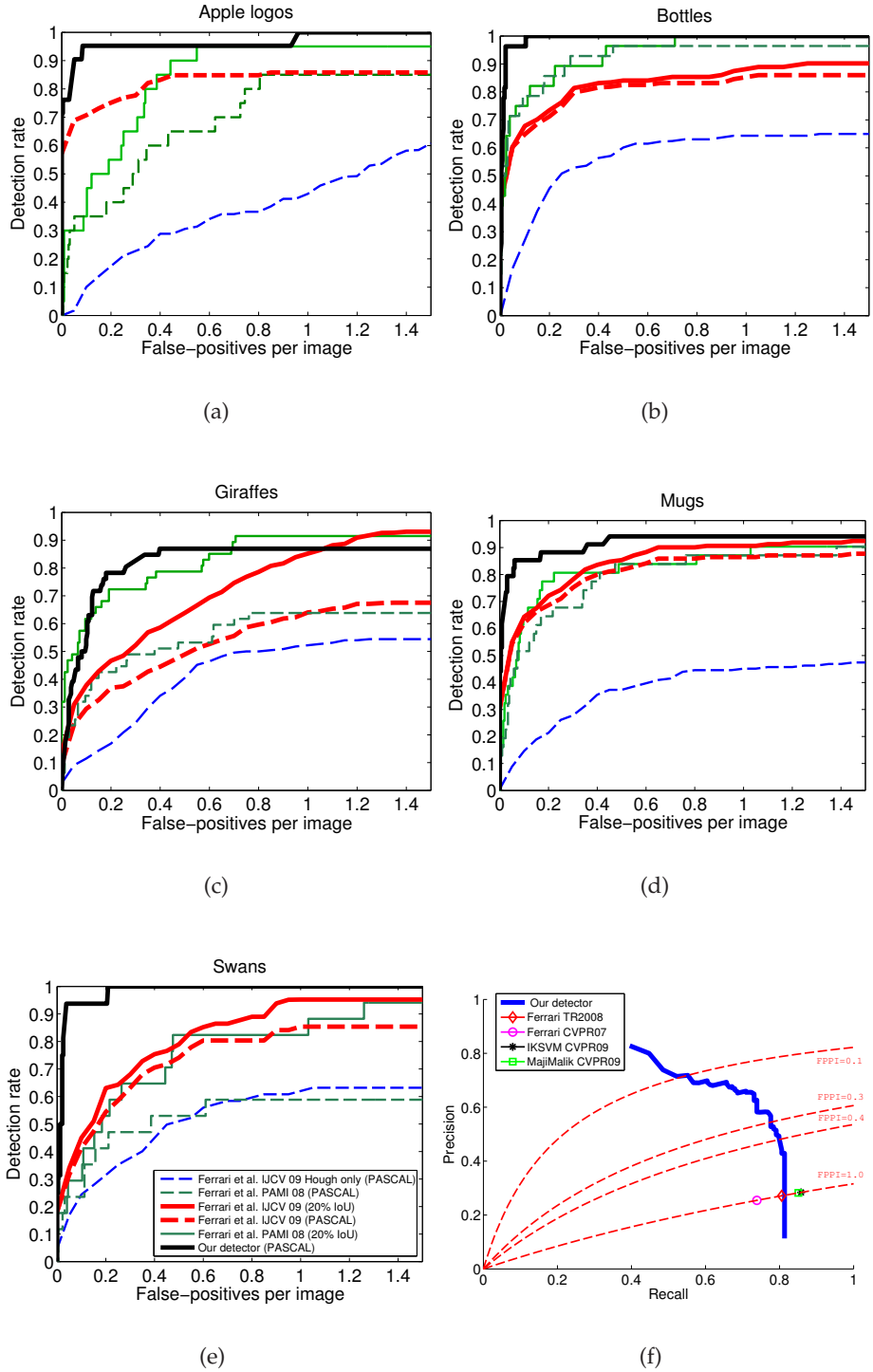


Figure 5.5: Detection performance for the ETHZ Shape classes (single best out of five runs) as detection/FPPI and INRIA Horses as precision/recall curve, both using 50%-PASCAL criterion.

5.5 Conclusion

In this chapter we have introduced a novel approach in the paradigm of contour-based object detection based on partial contour matches to a reference contour and show competitive results on state-of-the-art datasets like ETHZ Shape classes and INRIA Horses. In contrast to related work, we demonstrate how our novel contour matching relaxes the approximations by piecewise segments. By providing partial matches of contours we extending the search beyond local neighborhoods of interest points or figure/ground assignment for complete contours. Our system implicitly handles parts of a contour and thus does not require grouping long salient curves or harmful splitting of contours to be able to match parts. Though a verification stage is a vital part for a full object detection system, we believe the focus should lie on better reflecting the hypotheses voting space, since this has a direct effect on the speed and accuracy of the full detector performance.

The proposed method achieves excellent performance in the hypotheses voting despite employing less complex cost functions, category models or even rich appearance features as seen in related work. The main challenges however lies in two aspects. First, more complex interactions between contour fragments inspired by [63, 122, 133] are a powerful concept to provide stronger category models and grouping [87]. Second, learning implicit hierarchies [68] and discriminative weights [134, 182] for the contours will help distinguish the object contours from background clutter. We will investigate this in more detail in the next chapter.

Figure 5.6: Illustration of qualitative results for ETHZ Shape classes and INRIA Horses, showing the reference template used for partial matching and its matched contour parts in corresponding colors.

It is the shape of an airplane that enables it to fly.

Marius Leonardeau

6

Discriminative Fragments for Object Detection

In this chapter we propose a method for object category localization by jointly learning discriminative contour fragments and a model for the object category. Previous work, as described in Section 6.2, incorporates learning techniques only for object model generation or for verification after detection. We present an object detection method solely focusing on shape as underlying cue as opposed to rich appearance-based local features used in most approaches. In the learning phase, we interrelate local shape descriptions (fragments) of the object contour with the corresponding spatial location of the object centroid, described in Section 6.3. The local shape description exploits properties like distinctiveness, robustness and insensitivity to clutter and is well-suited for discriminative learning in a random forest learning scheme. For detection we hypothesize object locations in a generalized Hough voting scheme and the back-projected votes from the fragments allow to approximately delineate the object contour. We evaluate our method on the well-known ETHZ Shape classes and INRIA Horses datasets, where for both we achieve competitive results for the full framework and state-of-the-art performance for the hypotheses voting, as shown in Section 6.4.

Contents

6.1 Introduction

Object localization in cluttered images is a major challenge in computer vision. Most methods in this field learn an object category model, for example, from a set of labeled training images and use this model to localize previously unseen category instances in novel images. The final detection results are usually returned as bounding boxes, highlighting coarse instance locations and some methods also return object outlines.

In general, the model-based detection approaches can be divided into appearance- and contour-based methods. Appearance-based approaches first detect interest points and then extract powerful image patch descriptors from the local neighborhoods of the detected points using versatile features like color, texture or gradient information. In contrast, contour-based methods exhibit interesting properties like invariance to illumination changes or variations in color or texture. It is further well-established in the field of visual perception theory [23] that humans are able to identify specific objects despite only seeing a limited number of contour fragments, without considering any appearance information.

In this chapter we propose a method showing that it is sufficient to solely rely on shape cues for the task of object detection, i.e. we are deliberately neglecting appearance information. In particular, we determine discriminative contour fragments of an object category in combination with learning a category model. Hence, we show that the integration of local shape fragment descriptors into a generalized Hough voting scheme enables us to outperform previous methods on challenging reference datasets like the ETHZ Shape classes and INRIA Horses.

6.2 Related Work

There exists a vast history of related work for model-based category recognition and localization. In this section we will briefly review some of the basic graphical models used for learning the layout of an object category, and then compare state-of-the-art appearance- and contour-based methods for category learning. For a review of shape descriptors, we refer to the related work section in Chapter 3.

6.2.1 Graphical Models

A category model is a representation of a class of objects for describing the layout and interactions of the important parts of this category. For example, for a bicycle the important parts are the two wheels, the seat and the steering wheel. The layout now contains and constrains the position of the part to a feasible and valid configuration. This eliminates unlikely configurations and combines the individual local parts to a holistic representation of the entire object. A graphical model is commonly used to describe the connection between parts of an object. Such a graph is then defined as $G = (V, E)$, where V are the vertices or parts of the graphical model, and E are the edges or spatial connections between the parts. This formulation captures the layout and groups local parts to a more complete representation.

The range of models varies from nodes only over rich spatial relationships to rigid template models. A bag-of-words approach [187] is a loose model, where the parts are not connected at all and only the occurrence of parts is evaluated. The spatial relationships can be represented in multiple ways. The main differences are the representation of direction and loops in the graphical models. Markov Random Fields (MRF) [83] represent undirected models which focus more on correlation than causality. K-fan models [43] with on n nodes represent a family of graphical models that lie between the completely disconnected graph (k = 0) and the complete graph (k = n - 1). The k-fan model generalizes a single center star-graph to multiple center graphs. Star-graph models are essentially a tree connecting every part to a single center node [79, 121, 154]. This model is often used in connection with the generalized Hough transform [13] or center voting lines [152].

Richer connections of more spatial relationships are, for example, the constellation model [62]. A category is modeled by parts and the edges are modeled by spring costs, which allow a flexible movement of the parts while keeping the overall structure. In-

ference is however NP-hard when loops exist in the graph. Acyclic graphs are usually considered to overcome the computational costs. Such tree-shaped models are powerful to model kinematic structures of animate objects, where the nodes are parts and the edges are the joints connecting the parts. One example for such models are the pictorial structures [59, 60], where the inference is efficiently optimized by dynamic programming. More rigid models use a regular-spaced grid to form a template for objects used in a sliding window search [45, 119].

6.2.2 Appearance-based Approaches

Following is a review of examples of these graphical models dealing with appearance-based modeling. The most common approach in this field is the bag-of-visual-words model [187] where Sivic and Zisserman represent the object class as a collection of orderless local image descriptors [141]. In [119], Lazebnik et al. extended this approach by incorporating spatial arrangements of features in a pyramidal matching setup and showed improved detection performance. Another related appearance-based representative is the Implicit Shape Model (ISM) [120, 121]. Leibe et al. introduced the ISM as a class-specific codebook of local appearance features, where each descriptor additionally contains information about the relative object centroid location.

During recognition in test images, extracted image patches are matched to the codebook and cast probabilistic votes in terms of a generalized Hough transform [13] to hypothesize object locations. However, unsupervised integration of a large number of object parts in a generative codebook approach involves a large-scale clustering problem and a time-consuming matching step. Hence, research focused on replacing the generative codebooks by discriminative learning of the object model parts to overcome these problems.

In computer vision problems, the commonly used learning methods are Support Vector Machine (SVM), Boosting, and Decision Trees. Vapnik introduced the SVM [200] as a method to select a weighted subset of the training features to build a decision margin between two classes. Schapire proved that for boosting [77, 171] multiple weak classifier can be combined and weighted to create a strong classifier. Quinlan introduced the decision tree [160] as a set of decisions leading to a final classification, where the information gain determines the importance of the decisions. Recently random forests [32] have been introduced where an ensemble of decision trees are used for automatic training and testing. For example, random forests are frequently applied for the tasks of

image classification and semantic image segmentation as addressed in [145, 174, 184]. Gall and Lempitsky introduced an extension of the discriminative random forest framework, denoted as Hough forests [79, 151]. Instead of only learning the appearance of the model parts, they learn a discriminative combination for mapping image patch appearance and corresponding center votes.

Another approach incorporating the Hough transform in a discriminative learning approach was introduced by Maji and Malik [134]. They focus on finding object parts that are consistent in their locations and have high repeatability and assign them higher weights based on a maximum-margin learning for the voting step. Ommer and Malik [152] presented an extension for the Hough transform by extending conventional, spatial Hough voting with a dimension for scale. They analyze the voting space by agglomerative clustering of voting lines to deduce an initial set of detection hypotheses.

6.2.3 Contour-based Approaches

In this section, we review the main approaches addressing contour-based object category localization. The Active Shape Model (ASM) is a method by Cootes et al. [41], which models the shape variation using statistical point distributions of the boundary points. In [62], Fergus et al. showed a learning approach, which incorporates shape alongside appearance information for learning a joint spatial layout distribution in a Bayesian setting for a limited number of shape parts.

Ferrari et al. [66] build a contour segment network, which partitions image edges of the object model into groups of adjacent approximately straight contour segments. For matching, they find matching paths through the network of segments which resemble the outline of the modeled categories. In a later approach [63] they define the contour segments in groups of k adjacent segments (kAS) for learning a codebook of adjacent contours from cropped training images [64] in combination with Hough-based center voting and non-rigid thin-plate spline matching. Ravishankar et al. [162] also use short fragments, yet selecting slightly curved segments allows certain articulations by splitting edges at high curvature points, instead of approximating them by straight lines.

Shotton et al. [182, 183] and Opelt et al. [153] simultaneously introduced similar recognition frameworks based on selecting and boosting contour-based features. Both methods construct a codebook and employ local shape features in a star-graph around the object centroid. The contour fragments used in the codebook are selected by an AdaBoost [171] learning stage after clustering similar fragments. Each fragment is aware

of its position along the object contour and holds information about its spatial displacement to the object centroid. In the recognition phase, they compare contours learned from the training set to the edges found in the test images using clutter-sensitive chamfer matching and then cast center votes in a generalized Hough transformation to find object hypotheses. The main differences are that [182] use a single fragment per weak detector while [153] have a variable number. Further the centroid localization in [182] uses a discrete grid for vote accumulation and [153] employs mean shift for determining localization clusters.

In [122], Leordeanu et al. introduce a recognition system that models an object category using pairwise geometrical interactions of simple gradient features. The resulting category shape model is represented by a fully connected graph with its edges being an abstraction of the pairwise relationships. The detection task is formulated as a quadratic assignment problem. Lu et al. showed a method for grouping local interest points described by Shape Context [18] in a particle filter formulation [132] and evolved consistent models for detection under static observation. Bai et al. [12] captured and penalized intra-class shape variations within a certain bandwidth of the object by introducing a sliding window-style contour object model called Shapeband.

Zhu et al. [218, 220] formulated object detection as a many-to-many fragment matching problem. They utilize a strong perceptual contour grouping method to obtain long, salient edge contours. For finding matching candidates, these are compared using Shape Context descriptors. The large number of possible matchings is handled by encoding the shape descriptor in a linear form, where optimization can be done by linear programming. In an extending work [189, 190], Srinivasan et al. showed promising results for automatically obtaining object models from training data instead of using a single category shape prototype. However, their method also relies on the availability of long, salient contours and has high complexity with detection times in the range of minutes per image. In the previous chapter we performed object detection by partially matching detected edges to a prototype contour in the test images in combination with Hough-based center voting. In such a way, piecewise contour approximations and error-prone matches between coarse shape descriptions at local interest points are avoided.

A recent approach proposed by Yarlagadda et al. [214] aims on grouping mutually dependent object parts of uniformly sampled probabilistic edges. In their Hough voting stage, object detection is formulated as optimization problem which groups dependent parts, correspondences between parts and object models and votes from groups to ob-

ject hypotheses from the nearest neighbor matches without a priori training. Payet and Todorovic [159] proposed mining repetitive spatial configurations of contours in unlabeled images for training an object detector. The contours are represented as sequences of weighted beam angle histograms and are transferred into a graph of contours, whose maximum a posteriori assignment represents the shapes of discovered objects.

The approach of Amit and Geman [4] used randomized trees to perform shape recognition for handwritten digits by recursively partitioning the shape space and growing binary classification trees using geometric arrangements among local topographic codes as splitting rules. Fidler et al. [67, 69, 72] show a method for hierarchical statistical learning of local configurations of Gabor filters for category classification. This is extended to object detection by learning a hierarchical vocabulary for simple and complex parts [70, 71].

Another hierarchical approach by Kokkinos and Yuille [106] formulated the task as image parsing to provide fast coarse to fine matches. Other approaches employed game-theoretic grouping strategies such as dominant set clustering [157] to group shape-based part descriptions into coherent models [133, 210].

6.2.4 Summary

Recently, reasonable effort was placed onto fusing appearance- and contour-based methods like [123, 195]. Toshev et al. incorporate appearance information by using superpixel segmentations together with shape information derived from boundaries [195], and Li et al. exploit multiple overlapping figure-ground segmentations for an appearance-based recognition task [123]. The extensive related work stresses the importance of category model learning and exploitation of shape information for object detection.

However, in our observations we found that many approaches place an enormous effort into learning a suitable object model from training data [64, 153, 182], yet then neglect to apply this gained knowledge in a similar fashion during the matching phase. Instead, techniques like error-prone chamfer matching are used for inference whether an object is present or not.

In this chapter we propose a shape-based detection method, which benefits from inherently unifying the strengths of the strong shape-based edge description and discriminative learning for training and testing, instead of relying on separate techniques in concatenation. Hence, we show a method to jointly learn a novel shape fragment description with its spatial location information about the object contour. For recog-

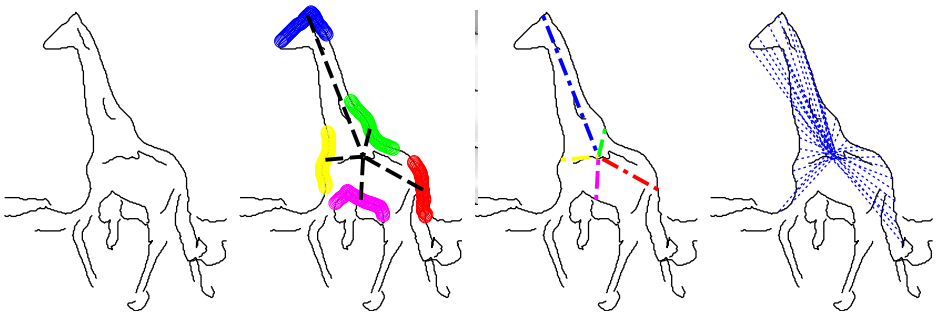
nition, we directly apply the learned category model in a generalized Hough voting manner. The overall goal is to simultaneously focus on object model generation and object detection within a single description, classification, and localization approach.

A key issue of such an approach is to use a powerful local shape descriptor, which fulfills the requirements of distinctiveness, robustness to clutter and noise, invariance and efficiency. Since most common local shape descriptors are limited concerning these requirements, we propose a novel fragment shape descriptor which describes relative spatial arrangements of sampled points by means of angular information. We employ the *Structured Measurement Descriptors* introduced in Chapter 3 and use a shape description especially suited for local fragment description. As it is shown in the experiments in Section 3.6, this descriptor outperforms related methods like Shape Context [19] for the classification scenario.

6.3 Discriminative Learning of Category Models

In this section we present our novel joint approach for shape-based category model learning and inference for object detection. The proposed method combines powerful local shape description and part knowledge with machine learning to derive a category shape model. This category model is implicitly used during the object detection phase to determine the location and outline of the underlying object instances.

The four main parts of our proposed object category localization method are illustrated in Figure 6.1. First, as underlying representation we use connected and linked edges as it is described in Section 6.3.1. Second, from the obtained edges we extract local fragment descriptions using a structured shape descriptor that captures local angular information along edges, as described in Section 6.3.2. Third, for the discriminative learning of fragments and their category model, we train from a set of labeled positive and negative training images. We use a random forest approach on the obtained descriptors and storing the relative location and scale of the fragments with respect to the object centroid for the positive training samples, as explained in Section 6.3.3. Fourth, at run-time, we cast probabilistic votes for possible center locations of the target objects in a generalized Hough voting manner. The resulting local maxima in the Hough space serve as detection hypotheses. For ranking and verification we process location hypotheses using a standard histogram of gradients (HOG) verification step, as outlined in Section 6.3.4.



(a) (b) (c) (d)

Figure 6.1: Illustration of our object category learning and localization: Local edge fragments are extracted and discriminatively trained in a random forest, which analyzes a triplet $\{\Psi_i, d_i, c_i\}$, where Ψ_i is our shape fragment descriptor, d_i its corresponding center voting vector and c_i its class label. During testing, the fragments are classified and cast a vote for object centroid to hypothesize the object locations in the image.

6.3.1 Edge Extraction

As a first step we extract edges of the input image. We use the Berkeley edge detector [137] in all experiments and link the binary image [109] into oriented, connected point coordinate lists of points. Please note the importance of perceptual linking [66, 219] which has great influence on the type of obtained fragments, since different splits of T-junctions or gap closing yield or exclude very different fragments. The obtained sequences of connected points are the basis for all subsequent steps. For this, we use the following terminology: A point is single point in a binary edge image and is defined as $\mathcal{P} = \{P_x, P_y\}$, where P_x and P_y are coordinates of the point. We define the linked sequences of points as *boundaries* $\mathcal{B} = (P_1, P_2, \ldots, P_N)$, where P_i is a single point and the number of points is N, which is arbitrarily long due to the underlying perceptual linking. Hence we denote all overlapping parts of a boundary as *fragments* $\mathcal{F} = (P_1, P_2, \ldots, P_K)$, where P_1 to P_K is a subset of the sequence of points in a boundary. In our proposed method all analyzed fragments have the same length, consisting of exactly K points, which are short yet highly overlapping fragments as opposed to long boundaries.

6.3.2 Fragment Description

For discriminative contour description and learning of local shape, we need a discriminative yet robust shape descriptor. These requirements are challenging as the edge extraction method only extracts very limited support windows for each fragment of constant size. Further we require a fast and repeatable way to describe the local shape. Previous related work on shape description either relies on very local features coupled with complex category models [20, 62, 122], or complex rich features [20, 45, 131].

In this chapter we focus solely on shape fragments as features and thus employ the Structural Measurement Description introduced in Chapter 3. Contrary to region or contour boundaries, fragments are very small parts of a contour and only deal with a very limited spatial extent. This brings various benefits, however also reduces the information that is captured by such local structures. The spatial extent now determines the size of the part. Contrary to a contour description, only a subset of points are available to derive a fragment description and the support window is given by the number of points surrounding the fragment in each direction.

For such short fragments it is especially valuable to maintain as much of the information as possible due to their limited local structure. Hence, we build a description Ψ in the following fashion encoding its orientation using a reference point on its bounding rectangle. An angle α_{ij} is calculated between a chord $\overline{P_iP_j}$ from a reference point P_i to another sampled point P_j and a chord $\overline{P_iP_*}$ from P_i to P_*, which is defined by the upper left corner of the bounding box of a fragment. In such a way, we define

$$\alpha_{ij} = \sphericalangle(\overline{P_iP_j}, \overline{P_iP_*}), \tag{6.1}$$

where P_* is a fixed reference point relative to the fragment. It is defined by the upper left corner of the bounding box of the boundary points for this fragment. This description encodes the limited spatial environment of each fragment. This limited support window leads to a compacter description as less points ($K \ll N$) are included compared to contour description. Partial description is achieved through redundant overlapping of such fragments. In this way a fixed length of K points results in a fixed descriptor size of K^2.

We are using a fragment-dependent reference point which is defined by the fragment's bounding box. Therefore it also contributes to a discriminative and local description of each considered fragment (see Figure 6.2). This reference sampling strategy is carefully chosen together with the discriminative learning framework and is crucial

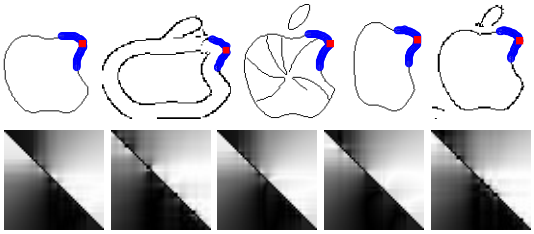


Figure 6.2: Fragment description for intra-class variations of similar part locations on the object.

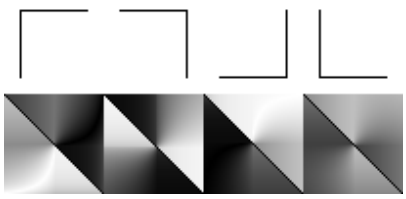


Figure 6.3: Selected shape primitives (first row) and their corresponding fragment description (second row).

for reasonable performance. Otherwise the learning algorithm would not be able to distinguish between different locations on regularly shaped object parts as for example the corner parts of a rectangle shown in Figure 6.3.

The three most important properties of this form of description are the increased distinctiveness, invariance and efficiency. First and most importantly, descriptors generated from fragments that are part of the foreground object need to be discriminative against those calculated from mere background clutter. The idea is to utilize the properties of the fragments within the Structured Measurement Descriptor to derive similar descriptors for similar parts of the object. We want the descriptors to be tolerate intra-class variations and small perturbations of the same fragments describing the same parts of the object. For different parts of the object, the fragment description should hence be distinguishable, as illustrated in Figures 6.3 and 6.2. The corner fragments of the rectangle each produce different descriptors, whereas the parts of the apple produce similar descriptors. The random forest learning framework is later employed to further boost this difference by learning discriminative weights.

Second, features are commonly classified according to their level of invariance to geometrical transformations. For example, features invariant to Euclidean transforms keep unchanged after applying translation and rotation. Increasing the degree of invariance generally decreases the distinctiveness and thus weakens the distinctiveness property. The fragment descriptor is invariant to translation and actively encodes orientation information, which increases the amount of information one can capture in such a limited local shape description.

Third, the feature description is efficiently computable by means of a lookup-table. We precompute the values for the angular measurements, hence reduce actual feature description to a constant-time lookup-operation. Since the angular description in Equa-

tion 6.1 corresponds to a surjective mapping $f : I \subset \mathbb{R}^2 \rightarrow [0, \pi]$, a lookup-table can be created for all possible fragment tuples $\{P_i, P_j\} \in I$ as follows. The maximum Euclidean distance between the reference point P_* and any fragment point P_i is bounded by the fragment length K. Hence, the area of the bounding box around the fragment corresponds to the total number of possible fragment point locations and defines I. Consequently, we can precompute a square matrix of size $|I| \times |I|$ holding the angular descriptions for all tuples $(P_i, P_j) \in I$. This matrix then serves as lookup table during feature calculation for arbitrary fragment points.

6.3.3 Discriminative Fragment Learning

For discriminatively learning our the fragment descriptor jointly with the category model, we use a formulation of the random forests [32] learning method. In addition to finding optimal splits of the training data, we use it to also optimize grouping of class-specific attributes like object center votes [79]. In such a way, a test sample is a local shape fragment extracted from the image, which is classified both according to its feature description as well as spatial configuration.

Random forests were initially introduced by Breiman [32] and are widely used for machine learning tasks in computer vision problems like recognition, visual tracking and classification. They inherently handle multi-class problems, are robust to certain amounts of label noise, and their performance is competitive to other learning methods. A random forest is a set of decision trees, where each node in the tree hierarchy contains a binary test, except for the final level of leaf nodes. For classification, a test sample is propagated on the way from the root to a single specific leaf node, depending on the binary test results in every intermediate node along its path. The training process of conventional random forests follows supervised and recursive construction of the trees by selecting binary split test to optimally separate class distributions. At the last level, the leaf nodes are used for storing information about the training sample distributions.

The binary tests are chosen randomly and the training procedure aims for splits of the input training samples to obtain disincentive class distributions, such that a Gini impurity or information gain is optimized. Recently, Gall and Lempitsky introduced an extension denoted as Hough forests [79], which in addition to finding optimal splits of the training data also optimizes grouping of class-specific object center votes. We employ this extension to learn and evaluate our shape fragment description and center vote vectors for category object localization.

Each tree is constructed on a set of training samples $\mathcal{S}_i = (\Psi_i, c_i, d_i)$, where Ψ_i is the fragment descriptor matrix, c_i its corresponding class label and d_i the center voting vector, describing the displacement to the center for positive training samples. Positive samples are selected by taking all edges within the ground truth bounding box annotations, and negative samples are selected from outside this bounding box. The binary test are chosen according to [79], randomly selecting to reduce the uncertainty in either class label or center offset information.

In every intermediate node, a binary test is selected for the training fragment descriptors Ψ_i. A test $t(\Psi_i) \rightarrow \{0,1\}$ is defined on two randomly chosen positions (p,q) and (r,s) in the $N \times N$ descriptor according to

$$t(\Psi) = \begin{cases} 0 \text{ if } \Psi_i(p,q) < \Psi_i(r,s) + \tau \\ 1 \text{ otherwise} \end{cases} , \tag{6.2}$$

where τ is a node-specific threshold value, that optimizes the data split in the intermediate nodes. Knowing the underlying features are the Structural Measurement Descriptors, the randomly chosen positions in the descriptor matrices are measurements about chord lines. A chord is a line joining two points on a shape fragment. Hence, the binary test compares two angles measured between two chords per angle. So contrary to simple pixelwise brightness comparison, these binary tests perform a local shape analysis of the spatial configuration of four chord lines. A large number of these randomized binary tests performed on training samples where the training labels c are known and allow to approximate the class distributions of the respective test data.

The 2D offset vectors d_i play an important role during the training of the category models and state the main difference to conventional random forests. Every positive training sample carries a directed voting vector from its fragment center point P_c to the center of the object bounding box P_{bbox}. The voting vectors for negative samples are defined as the origin (0,0). After training is complete, the ratio between positive and negative samples p_c, and a list of all center offset vectors $D_L = \{d_i\}$ are stored in each of the leaf nodes.

Similar to the construction of random forests, the trees are recursively built from the root node and the training data is split in the intermediate nodes into two sets according to an optimality test. For the optimality test, we follow the approach in [79], where both, the class ratios p_c and the offset lists D_L are used in the optimality tests to reduce the decision uncertainties towards the leaf nodes in each tree. The quality of the binary

tests, applied to a subset of samples $S_{n_j} = \{\mathcal{S}_i = (\Psi_i, c_i, d_i)\}$ available in the current node n_j, is evaluated either by the class-label uncertainty or the offset uncertainty. The class-label uncertainty is defined as

$$U_1(S) = |S| \cdot H(c_i) \tag{6.3}$$

where $|S|$ is the size of the current set of training samples and H is the binary entropy of the class labels c_i, which is defined as

$$H(c_i) = p_c \cdot \log p_c - (1 - p_c) \cdot \log(1 - p_c) \tag{6.4}$$

where p_c is the probability of the class ratios stored in the current node. This measure defines the impurity of all class labels $\{c_i\}$ reaching this node.

The second measure evaluates the center voting offset and thus essentially the spatial configuration of the parts. The offset uncertainty is defined as

$$U_2(S) = \sum_{\forall i | c_i = 1} (d_i - d_{mean}) \tag{6.5}$$

where d_{mean} is the mean offset vector over all offset vectors contained in training sample set S_{n_j} at this node. This measure describes the scatter of all offset vectors reaching the current node. The goal is to minimize the scatter and thus group together similar spatial configurations.

For a split, we define three parameters, which determine the binary split of the training data. A set of position tests $\{t^k\}$ is selected with uniformly sampled locations (p, q, r, s). Further a random decision $x = \{1 \vee 2\}$ determines whether the class-label or the offset uncertainty is optimized. The test-specific threshold value τ^k is chosen at random in the range of real-valued differences for a test t^k. Over all selected tests in $\{t^k\}$, we choose the split for which the training data is best minimized by the selected uncertainty measure according to

$$\min_k \Big(U_x(\{p_i | t^k(\Psi_i) = 0\}) + U_x(\{p_i | t^k(\Psi_i) = 1\}) \Big) \ . \tag{6.6}$$

As a result of randomly selecting between both criteria $U_1(S)$ and $U_2(S)$, the classification and voting direction uncertainties for a given test sample alternatingly decrease on its way through a tree.

Once the entire forest built from multiple decision trees is constructed, the implicity category model is trained and the detection process is ready to start evaluation on the test images. Edges are extracted from the test images and arranged into ordered, connected edge lists $\mathcal{B}$ with $|\mathcal{B}| = N \geq K$. For each $\mathcal{B}$, again a total number of $(|\mathcal{B}| - N + 1)$ overlapping fragment descriptors $\{\Psi_j\}$ are computed, and then classified down the trees into leaf nodes. Please note, the descriptors are only computed along edges, which significantly reduces the computational costs in comparison to a sliding window approach or dense sampling as used in [79].

Since the voting vectors D_L and class label probabilities p_c are known in every leaf node, we are able to cast the voting vectors into the Hough space V and accumulate the voting probabilities. Finally, the resulting Hough image is Gauss-filtered and its local maxima hypothesize the detected object locations. Back-projecting the votes collected in a location hypotheses allows to approximately delineate the object outline, as illustrated for several test images and category models in Section 6.4.

6.3.4 Ranking and Verification

The previous stage of our method provides object hypotheses in the test image and a corresponding score obtained from the Hough voting space. For verification of these hypotheses, we additionally provide a ranking employing a pyramid matching kernel (PMK) [89]. Since our hypotheses generation stage delivers only a small set of hypotheses per image, a local verification is still efficient. Opposed to sliding window approaches, an order of magnitude fewer candidates are considered as hypotheses.

The PMK classifier is trained on Histograms of Oriented Gradient (HOG) features [45] from the same training examples selected also for training the category models in the previous section. We use the classifier for ranking and for verification, where additional nearby locations and scales around the proposed hypotheses are searched, similar to the related work [134, 152, 164].

6.4 Experimental Evaluation

In order to demonstrate the benefits of our proposed method for discriminative shape and category model learning, we show the performance of our method on challenging datasets like the ETHZ Shape classes [66] and the INRIA Horses [65, 101]. As our proposed approach is contour-based and does not exploit color appearance or additional segmentations as in [123, 195], we compare our results to several related contour-based recognition approaches.

For all our experiments, we use the following parameters. The random forest classifier consists of 12 decision trees, each with a depth of 15. The fragment length is fixed to $N = 51$, as the classification task in the Chapter 3 suggested is a good balance between recall and discrimination. We randomly extract 10 000 positive training samples from edges within the bounding box annotation. The positive training images are all scaled to the median height of the selected training dataset and the aspect ratio is fixed. 10 000 negative training samples are extracted from the same training images, yet outside of the bounding boxes. Detection performance is evaluated using the strict PASCAL-50% criterion, which requires that the intersection of the bounding box of the object hypotheses and the ground truth over the union of the two bounding boxes is larger than 50%.

Since our method is not implicitly scale invariant, we run the detector on multiple scales. However, this is efficiently done since descriptor calculation takes constant time when using the lookup tables and traversing the trees has logarithmic complexity. In practice, the average evaluation time is a few seconds per image for our C implementation, while the dense sampling used in [79] take minutes to evaluate all scales.

6.4.1 ETHZ Shape Classes

The ETHZ Shape dataset consists of five object classes (applelogos, bottles, giraffes, mugs and swans) and a total of 255 images. The images contain at least one and often multiple instances of a class instance and have a large amount of background clutter. All classes contain significant intra-class variations and scale changes. Therefore, we run the detector on 15 different scales, equally distributed between factors of 0.2 and 1.6. We use the same test protocol as specified in [65], where a class model is learned by training on half of the positive examples from a class, while testing is done on all remaining images from the entire dataset.

Figure 6.4: Examples of back-projected contour fragments for a detected object hypothesis for all classes of ETHZ. The back-projection allows to approximately delineate the object outline (intensity of the outline indicates the weight).

The hypotheses locations are ranked according to their confidence scores. In the voting stage this confidence corresponds to the accumulated values in the Hough voting space. We provide a ranking according to a pyramid matching kernel (PMK) [89]. The PMK classifier is trained on the same half of the positive example from the training set and an equal number of windows sampled from negative training images. This is according to the verification step as proposed by Maji and Malik [134]. We use the classifier for ranking and for verification, where we also sample nearby locations and scales in a small grid around the proposed hypotheses locations. Including the local search is still very efficient since our hypothesis stage delivers on average 3.5 bounding boxes per image, which is orders of magnitudes lower than a sliding window detectors require [90]. Figure 6.4 further illustrates several examples of cropped bounding box hypotheses showing the back-projected fragments into the original image locations.

In Table 6.1 we compare the results of our described object detector for each object class in relation to current state of the art [134, 152, 164, 214], where divisions into voting, ranking and verification stages are applicable. However, due to the large number of competing methods, we only provide the scores of the initial Hough *voting* stage and the PMK *ranking* stage evaluated at 1.0 FPPI. Recognition performance is evaluated by ranking the hypotheses according to their confidence scores. Finally, we also show results of our method for the full verification step (where also nearby locations and scales are tested around the returned hypotheses) at FPPI $= 0.3/0.4$, which is the standard measure for comparing results on the ETHZ Shape classes dataset.

	Voting Stage (FPPI=1.0)							*Ranking Stage (FPPI=1.0)*			*(FPPI=0.3/0.4)*
ETHZ Classes	Hough [65]	w_{ac} [152]	M^2HT [134]	PC [164]	Group [214]	Hough Forest [79]	**Our work**	[152] + PMK	[164] + PMK	**Ours PMK**	**Our work Verification**
Apples	43.0	80.0	85.0	90.4	84.0	80.0	94.4	80.0	90.4	100.0	94.4/100
Bottles	64.4	92.4	67.0	84.4	93.1	70.8	90.9	89.3	96.4	95.5	100/100
Giraffes	52.2	36.2	55.0	50.0	79.5	60.5	86.7	80.9	78.8	93.3	91.1/93.3
Mugs	45.1	47.5	55.0	32.3	67.0	73.1	92.3	74.2	61.4	88.5	80.8/87.2
Swans	62.0	58.8	42.5	90.1	76.6	81.3	73.3	68.6	88.6	93.3	100/100
Average	53.3	63.0	60.9	69.4	80.0	73.1	**87.5**	78.6	83.2	**94.2**	**93.3/96.1**

Table 6.1: Hypothesis voting and ranking showing detection rates (measured using PASCAL-50% criterion) for the ETHZ Shape classes [66]. For the voting stage our coverage score increases the performance by 7.5% [214], 18.1% [164], 24.5% [134], 26.6% [152] and 34.2% [65]. After ranking the hypotheses, we achieve an improvement of 11.0% over [164] and even 15.6% over [152].

As shown in Table 6.1, we substantially outperform the currently best scoring methods for both, Hough-voting and ranking stage. We achieve a performance boost of 7.5% over the previously best voting method in [214], 18.1% over [164], 26.6% over [152], 24.5% over [134] and even 34.2% over [65]. After applying the learned HOG models for ranking we are 11.0% better than the previously best scoring method [164] and 15.6% better than [152]. Finally, our method also shows high performance for the full verification system, providing an average recognition score of 93.3/96.1% at 0.3/0.4 FPPI, which is approximately on par with the highest scores reported from contour-based approaches in [190] (95.2/95.6%). However, the authors in [190] do not provide scores for detection and refinement stages individually which makes direct comparison difficult. Furthermore, their method has a high computational complexity and detection takes minutes per image. Further recent work incorporates appearance-based color segmentations to perceptually group the shape information in the work [195] (94.3/96.0%) and in [123] (average precision of 90.25% at 0.02 FPPI).

To further emphasize the contribution of our proposed shape descriptor with respect to the original Hough forest framework, we trained and evaluated on the same number of trees using all 32 provided features of [79], which are mostly pixelwise appearance information including color channels, gradients and small HOG-like features. To have a fair comparison and accomplish each of the 15 individually considered scales in reasonable time, we evaluated on the same locations as we did for our descriptor. As shown in Table 6.1, our single feature descriptor clearly outperforms the standard Hough forest method using all 32 features (14.4% better).

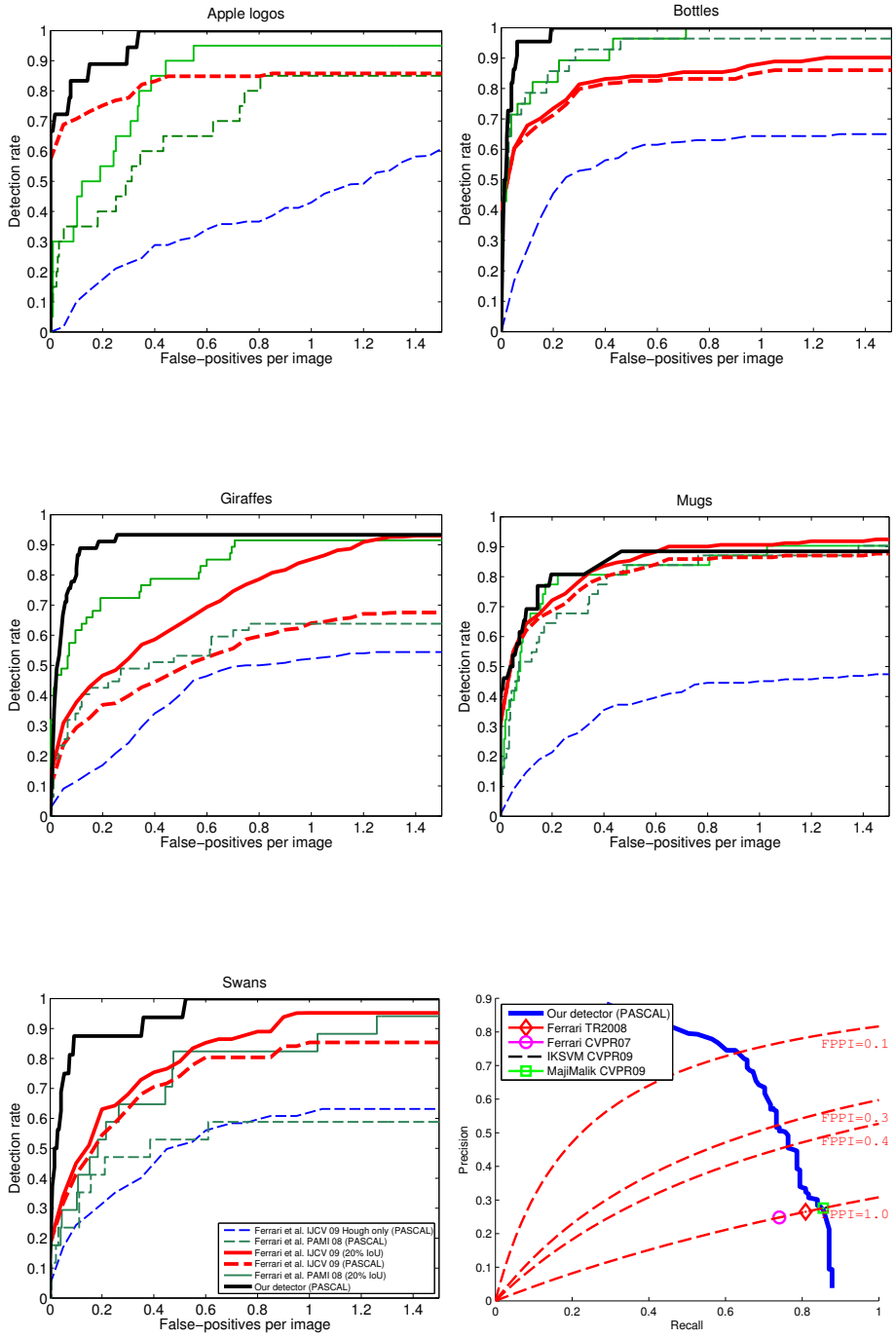


Figure 6.5: Object detection performance on the ETHZ Shape classes and INRIA Horse datasets. Each plot shows curves for the 50%-PASCAL and 20%-IOU criterion of the methods. Our results are shown in thick solid black/blue curves. Note, that we outperform the related work [63, 65] consistently over all classes.

In Figure 6.5 we show the detection rate vs. FPPI plots for all ETHZ Shape classes in comparison to all results provided by Ferrari et al. [63, 65] and also the recent results on INRIA Horses dataset. Figure 6.6 illustrates some detection results for different classes. We show the highly cluttered edge responses of the test images which are used for localization and the corresponding back-projections of the classified fragments for an object hypothesis which enables to additionally approximately delineate the object contour. The circles indicate the voted center locations of the hypotheses. The degree of intensity along the edges corresponds to the weights of the extracted fragment around this point, where darker is more important.

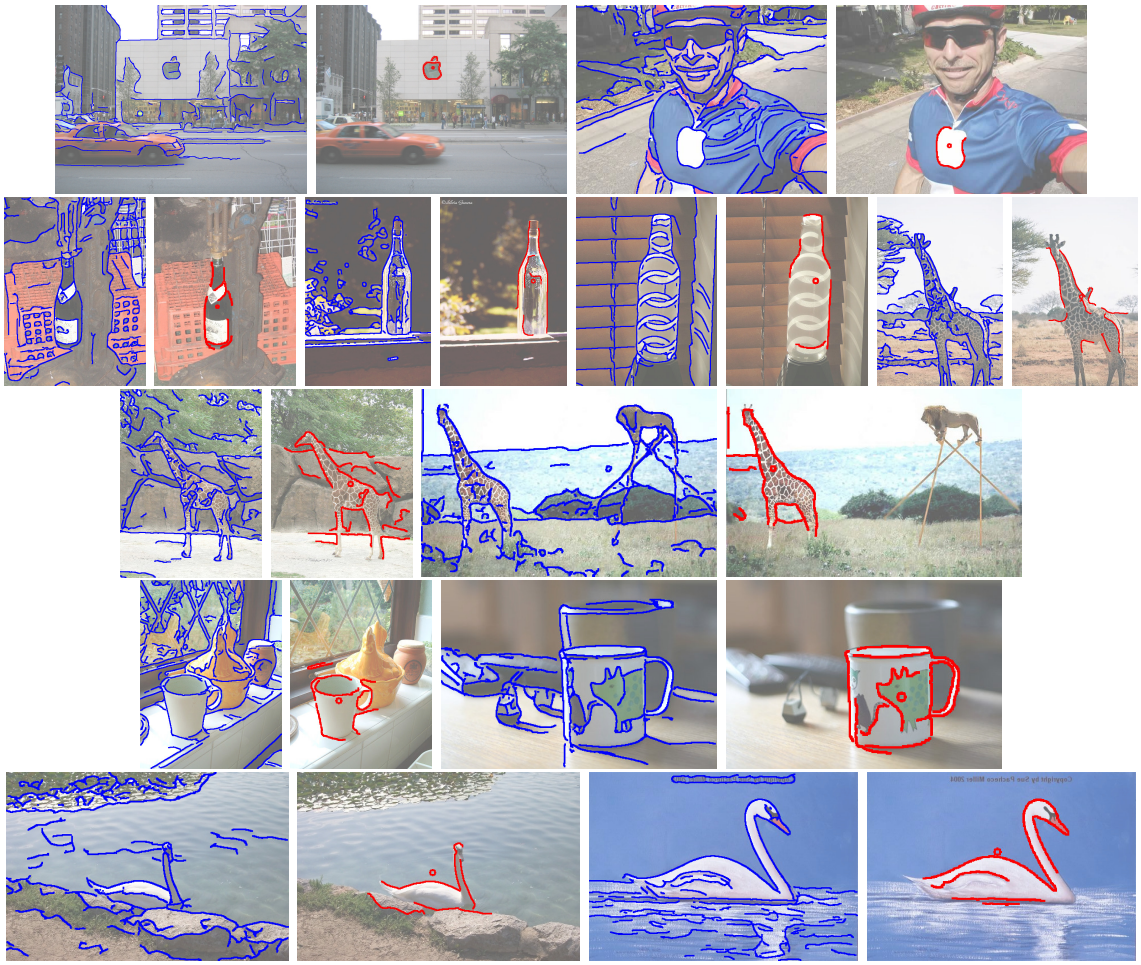


Figure 6.6: Examples for successful object localizations for all classes of ETHZ Shape classes [66]. The cluttered edge responses (in blue) and the back-projected fragments (in red) for the object hypothesis with the highest confidence per image are shown.

INRIA Horses: *Final detection (FPPI = 1.0)*					
M^2HT [134]	Group [214]	PC [164]	PAS [63]	KAS [64]	**Our work** [108]
85.30	87.3	83.72	80.77	73.75	**85.50**

Table 6.2: Detection performance for discriminative shape fragment-based object detection for the INRIA Horses [101] reported at 1.0 FPPI using 50%-PASCAL.

6.4.2 INRIA Horses

The INRIA Horses dataset [65, 101] contains a total number of 340 images where 170 images belong to the positive class showing at least one horse in side-view at several scales and 170 images without horses as background images. The experimental setup is chosen as in [65], where the first 50 positive examples are used for training and the rest of the images are used for evaluation (120+170). We run the detector on eight different scale factors between 0.5 and 1.5. As shown in Table 6.2 compared to recently published results, we achieve a competitive detection performance of **85.50%** at 1.0 FPPI.

6.5 Conclusion

In this chapter we investigated the use of shape description and discriminative category model learning for the task of object detection. Our novel method discriminatively learns local shape fragment descriptors in combination with their spatial location relative to the object center in a random forest classifier to build a category object model. The design of the fragment descriptor abstracts connected edge image points into angular relations and learns their spatial configuration at a local descriptor level as well as for the category model layout. As demonstrated the proposed descriptor shows distinctive patterns for differently shaped fragment primitives, while allowing to tolerate small perturbations and intra-class variations. The experimental evaluation shows excellent results on the well-known ETHZ Shape and INRIA Horses datasets. Our method outperforms the currently highest scoring contour-based methods by approximately 8% at 1.0 FPPI after the Hough voting stage at reduced runtime of only a few seconds per image. In addition, we demonstrated that back-projections of the shape fragments voting for a hypothesis allows delineating the object outline. Future work will focus on turning back-projection information into concise image segmentations.

What we need are a few crazy people;
look at what we've achieved with the normal ones.

George Bernard Shaw (1856-1950)

7

Joining Classification, Localization and Segmentation

In this chapter we propose a method which jointly optimizes over object detection and segmentation. These are usually considered as independent consequent steps, as discussed in Section 7.2. Our method is attached to any available generalized Hough voting method and implicitly provides detection hypotheses and corresponding segmentations. In Section 7.3 we describe our method denoted as Hough Regions formulating the problem of Hough space analysis as Bayesian labeling of a random field exploiting provided classifier responses, object center votes and low-level cues like color consistency, which are combined into a global energy term. We propose a greedy approach to solve this energy minimization problem providing a pixel-wise assignment of each pixel in a test image as background or as belonging to a specific category instance. In such a way we bypass the parameter sensitive non-maximum suppression that is required in related methods. Experimental evaluation in Section 7.4 demonstrates that state-of-the-art detection and segmentation results are achieved and that our method is able to inherently handle overlapping instances, larger range of articulations, aspect ratios and scales in comparison to related approaches on multiple instance localization.

Contents

7.1 Introduction

Detecting instances of object categories in cluttered scenes is one of the main challenges in computer vision. Standard recognition systems define this task as localizing category instances in test images up to a bounding box representation, which limits overlapping instances due to the ambiguous assignment within overlapping bounding boxes. In contrast, semantic segmentation methods try to accurately outline the object of interest. Inherently this is the same task, as the segmentation of an object delivers the localization and the assignment of pixels which belong to the object. This assignment makes the recognition part easier, since misleading background information is discarded.

Both tasks have recently enjoyed vast improvements and we are now able to accurately recognize, localize and segment objects in separate stages using complex processing pipelines. Problems still arise where the recognition is confused by background clutter or occlusions, localization is confused by proximate features of other classes, and segmentation is confused by ambiguous assignments to foreground and background.

In this chapter we propose an object detection method which combines instance localization with segmentation in an implicit manner for jointly finding optimal solutions. Related work in this field, as will be discussed in detail in the next section, tries to combine detection and segmentation in separate subsequent stages [161], mostly considering objects independently by a crude separation using bounding boxes. In contrast, full scene understanding jointly estimates a complete scene segmentation of an image for all object categories [114], which is not the goal in this chapter. However, our instance segmentations could be used as input for the higher-order potentials in [114].

We place our method in the middle and combine localization and segmentation jointly by considering individual object instance localization and their segmentation iteratively. We achieve the instance localization by the extraction of so-called *Hough regions*,

(a) (b) (c) (d)

Figure 7.1: *Hough regions* (a) find reasonable instance hypotheses (b) despite strongly smeared vote maps. By jointly optimizing over both segmentation and detection, we identify even strongly overlapping instances (c) compared to standard non maximum suppression (d), which fails (red BB) to identify all instances.

exploiting available information from a generalized Hough voting method as illustrated in Figure 7.1. *Hough regions* are an alternative to the de-facto standards of bounding box non-maximum suppression or vote clustering, as they are directly optimized on the Hough vote space. Since we are considering maxima regions instead of single maxima, articulation and scale become far less important. Our method is thus more tolerant against diverging scales and the number of overall scales in testing is reduced. Further, we outline the object instances and achieve better recall for overlapping instances.

7.2 Related Work

In this chapter we consider the problem of localizing category instances in images and additionally provide accurate segmentations per instance. For providing such results mainly three different related research fields should be considered: non-maximum suppression, segmentation and scene understanding.

One of the most prominent approaches for improving object detection performance is non-maximum suppression (NMS) and its variants. NMS aims for suppressing all hypotheses (i. e. bounding boxes) within a certain distance (e. g. the widespread 50% PASCAL criterion) and localization certainty with respect to each other. Barinova et al. [14, 15] views the Hough voting step as an iterative procedure, where each bounding box of

an object instance is greedily considered. Desai and Ramaman [46, 47] see the bounding box suppression as a problem of context evaluation. In their work they learn pairwise context features, which determine the acceptable bounding box overlap per category. For example, a couch may overlap with a person, yet not with an airplane.

The second approach for improving object localization is to use the support of segmentations. The work of Leibe et al. [121] introduced an Implicit Shape Model (ISM) which captures the structure of an object in a generalized Hough voting manner. They additionally provide segmentations per detected category instance, but require ground truth segmentations for every positive training sample. To recover from overlapping detections, they introduce a minimum description length (MDL) criterion to combine detection hypotheses based on their costs as separate or grouped hypotheses. Borenstein and Ullman [28] generate class-specific segmentations by a combination of object patches, yet this approach is decoupled from the recognition process. Yu and Shi [215] show a parallel segmentation and recognition system in a graph theoretic framework, but are limited to a set of a priori known objects. Amit et al. [5] are treating parts as competing interpretations of the same object instances. Larlus and Jurie [115] showed how to combine appearance models in a Markov Random Field (MRF) setup for category level object segmentation. They used detection results to perform segmentation in the areas of hypothesized object locations. Such an approach implicitly assumes that the final detection bounding box contains the object-of-interest and cannot recover from examples not sticking to this assumption, which is also the case for methods in full scene understanding. Gu et al. [90] use regions as underlying reasoning for object detection, however rely on a single over-segmentation of the image, which cannot recover from initial segmentation errors. Tu et al. [196] propose the unification of segmentation, detection and recognition in a Bayesian inference framework where bottom-up grouping and top-down recognition are combined for text and faces.

The third approach for improving localization is to strive for a full scene understanding to explain every object instance and all segmentations in an image. Gould et al. [86] jointly estimate detection and segmentation in a unified optimization framework, however with an approximation of the inference step, since their cost formulation is intractable otherwise. Such an approach cannot find the global optimum solution. Wojek and Schiele [207] couple scene and detector information, but due to the inherent complexity the problem is not solvable in an exact manner. Winn and Shotton [205] propose a layout consistent Conditional Random Field (CRF) which splits the object into parts

and enforces a coherent layout when assigning the labels and connects each Hough transform with a part to extract multiple objects. Yang et al. [211, 212] propose a layered object model for image segmentation, which defines shape masks and explain the appearance, depth ordering, and labels of all pixels in an image.

Ladicky et al. [112, 114] combine semantic segmentation and object detection into a framework for full scene understanding. Their method trains various classifiers for *"stuff"* and *"things"* and incorporates them into a coherent cooperating scene explanation. So far, only Ladicky et al. managed to incorporate information about object instances, their location and spatial extent, as important cues for a complete scene understanding, which allows to answer a question like what, where and how many object instances can be found in an image. However, their system is designed for difficult full scene understanding and not for an object detection task, as they only integrate detector responses and infer from the corresponding bounding boxes. This limits the ability to increase the detection recall or to improve the accuracy of instance localization.

Our method improves the accuracy of object detectors by using the object's supporting features for joint segmentation and reasoning between object instances. We achieve a performance increase in recall and precision, through the ability to better handle articulations and diverging scales. Additionally, we do not require a parameter sensitive non-maximum suppression since our method delivers unique segmentations per object instance. Please note, in contrast to related methods [74, 121] these segmentation are provided without learning from ground truth segmentation masks for each category.

7.3 Joint Reasoning of Instances and Support

The main goal of this chapter is the joint reasoning about object instances and their segmentations. Our starting point is any generalized Hough voting method like the Implicit Shape Model (ISM) [121], the Hough Forests [79] or the max-margin Hough transform [134]. We formulate our problem in terms of a Bayesian labeling of a first-order Conditional Random Field (CRF) aiming at the minimization of a global energy term. Since global inference in the required full random field is intractable, we propose a greedy method which couples two stages iteratively solving each stage in a global optimal manner. Our model inherently links classifier probabilities, corresponding Hough votes and low-level cues such as color consistency and centroid proximity.

We first describe in Section 7.3.1 the global energy term for providing pixel-wise assignments to category instances, analyzing unary and pairwise potentials. In Sec-

tion 7.3.2 we introduce our approach for minimizing the energy term by a greedy approach. Finally, in Section 7.3.3 we directly compare the properties of our method to related approaches in this field.

7.3.1 Probabilistic Global Energy

We assume that we are given any generalized Hough voting method like [79, 121, 134, 151], which provides our N voting elements X_i within the test image and corresponding object center votes H_i into the Hough space. We further assume that we are given $p(C|X_i, D_i)$ per voting element, which measures the likelihood that a voting element X_i belongs to category C by analyzing a local descriptor D_i, for example feature channel differences between randomly drawn pixel pairs as in [79]. All this information can be directly obtained from the generalized Hough methods, see experimental section for implementation details.

The goal of our method is to use the provided data of the generalized Hough voting method to explain a test image by classifying each pixel as background or as belonging to a specific instance of an object category. We formulate this problem as Bayesian labeling of a first-order Conditional Random Field (CRF) by minimizing a global energy term to derive an optimal labelling.

Such random field formulations are widespread, especially in the related field of semantic segmentation, e. g. [184]. In a standard random field $\mathcal{X}$ each pixel is represented as a random variable $X_i \in \mathcal{X}$, which takes a label from the set $\mathcal{L} = \{l_1, l_2, \ldots, l_k\}$ indicating one of k pre-defined object classes as e. g. grass, people, cars, etc. Additionally, an arbitrary neighborhood relationship $\mathcal{N}$ is provided, which defines cliques c. A clique c is a subset of the random variables $\mathbf{X}_c$, where for $\forall i, j \in c$ holds $i \in \mathcal{N}_j$ and $j \in \mathcal{N}_i$, i. e. they are mutually neighboring concerning the defined neighborhood system $\mathcal{N}$.

In our case, we not only want to assign each pixel to an object category, we additionally aim at providing a unique assignment to a specific instance of a category in the image, which is a difficult problem if category instances are highly overlapping. For the sake of simplicity, we define our method for the single class case, but the method is easily extendable to a multi-class setting.

We also represent each pixel as random variable X_i with $i = 1 \ldots N$ and aim at assigning each pixel a category-specific *instance* label from the set $\mathcal{L} = \{l_0, l_1, \ldots, l_L\}$, where L is the number of instances. We use label l_0 for assigning a pixel to the back-

ground. We seek for the optimal *labeling* $\mathbf{x}$ of the image which is taken from the set $\mathbf{L} = \mathcal{L}^N$. The optimal labeling minimizes the general energy term $E(\mathbf{x})$ defined as

$$E(\mathbf{x}) = -log\ P(\mathbf{x}|\mathbf{D}) - log\ Z = \sum_{c \in \mathcal{C}} \psi_c(\mathbf{x}_c), \tag{7.1}$$

where Z is a normalization constant, $\mathbf{D}$ is the observed data, $\mathcal{C}$ is the set of considered cliques defined by the neighborhood relationship $\mathcal{N}$ and $\psi_c(\mathbf{x}_c)$ is a potential function of the defined clique. The Maximum a Posteriori (MAP) labeling $\mathbf{x}^*$ is then found by minimizing this energy term as

$$x^* = \arg\min_{x \in \mathbf{L}} E(\mathbf{x}). \tag{7.2}$$

To obtain reasonable label assignments it is important to select powerful potentials, which range from simple unary terms (e. g. evaluating class likelihoods), pairwise potentials (e. g. the widespread Potts model) or even higher order potentials. While unary potentials ensure correct label assignments, pairwise potentials aim at providing a smooth label map, e. g. by avoiding the assignment of identical labels over high image gradients. The overall goal is to provide a smooth labeling consistent with the underlying data.

Our energy is modeled as the sum of unary and pairwise potentials. In the following, we assume that we are given a set of seed variables $\mathcal{S}$ which consist of L subsets $\mathbf{S_l} \in \mathcal{X}$ of random variables for which we know the assignment to a specific label in the image, i. e. we know the instance it belongs to. We again assume that the set $\mathbf{S_0}$ represents the background. These assignments are an essential step of our method and how to obtain them is discussed in Section 7.3.2.

Based on the given assignments $\mathbf{S_l}$, we define our instance-labeling problem as minimizing the following energy term

$$E(x) = \sum_{X_i \in \mathcal{S}_l} \Theta(X_i) + \sum_{X_i \in \mathcal{S}_l} \Upsilon(X_i) + \sum_{X_i \in \mathcal{S}_l} \Omega(X_i) + \sum_{X_i, X_j \in \mathcal{N}} \Psi(X_i, X_j), \tag{7.3}$$

which contains a unary, instance-specific class potential Θ, a color based unary potential Υ, a distance based unary potential Ω and a pairwise potential Ψ. The first unary potential Θ contains the estimated likelihoods for a pixel taking a certain category label as provided by the generalized Hough voting method as

$$\Theta = -\log\ p(C|X_i, D_i), \tag{7.4}$$

i. e. the unary potentials Θ describe the likelihood of getting assigned to one of the identified seed regions or the background. Since we do not have a background likelihood, the corresponding values for l_0 are set to a fixed constant value p_{back}, which defines the minimum class likelihood we want to have. The term Θ drives our label assignments to correctly distinguish background from actual object hypotheses.

The unary potential Υ analyzes the likelihood for assigning a pixel to one of the defined seed regions, e. g analyzing local appearance information in comparison to the seed region. In general any kind of modeling scheme is applicable in this step. We model each subset $\mathbf{S_l} \in \mathcal{X}$ by Gaussian Mixture Models (GMM) $\mathcal{G}_l$ and define color potentials for assigning a pixel to each instance by

$$\Upsilon = -\log\ p\left(X_i|\mathcal{G}_l\right). \tag{7.5}$$

The corresponding likelihoods for the background class l_0 are set to $1 - \max_{l\in\mathcal{L}} \log\ p\left(X_i|\mathcal{G}_l\right)$ since the background is mostly too complex to model. The term Υ ensures that pixels are assigned to the right instances by considering the appearance of each instance.

The third unary potential Ω defines a spatial distance function analyzing how close each pixel is to each seed region $\mathbf{S_l}$ by

$$\Omega = \Delta\left(X_i, \mathbf{S_l}\right), \tag{7.6}$$

where Δ is a distance function. The term Ω ensures that correct assignments to instances are made considering constraints like far away pixels with diverging center votes are not assigned to the same instance.

Finally, our pairwise potentials encourage neighboring pixels in the image to take the same label considering a standard Potts model Ψ analyzing pixels included in the same cliques c defined as

$$\Psi = \begin{cases} 0 & \text{if} \quad x_i = x_j \\ K & \text{if} \quad x_i \neq x_j \end{cases}, \tag{7.7}$$

where K is an image specific gradient measure on local color difference. The term Ψ mainly ensures that smooth label assignments are achieved. In such a way we can effectively incorporate not only the probability of a single pixel belonging to an object, but also consider the spatial extent and the local appearance of connected pixels in our

inference. Please note, that this formulation can be extended to superpixels [201] to infer coherent groups [113] or include higher-order potentials [105].

Finding an optimal solution without knowledge of the seeds $\mathbf{S_l}$ is infeasible since inference in such a dense graph is NP-hard. Therefore, in the next section we propose a greedy algorithm for solving the above presented energy minimization problem, which alternately finds optimal seed assignments and then segments instances analyzing the hough vector support. The final result of our method is a segmentation of the entire image into background and individual category instances.

7.3.2 Instance Labeling Inference

We propose a novel, greedy inference concept to solve the energy minimization problem defined in Equation 7.3. The core idea is to alternately find a single, optimal seed region analyzing the provided Hough space (Eq. 7.8), and to afterwards use this seed region to find an optimal segmentation of the corresponding instance in the image space (Eq. 7.3). The obtained segmentation is then used to update the Hough space information (Eq. 7.10), and in such a way we greedily obtain our final image labeling.

We formulate our instance labeling problem as a conventional Conditional Random Field (CRF). Due to the known votes $\mathcal{H}_i$ of each pixel into the Hough vote map, we have a connection between pixelwise feature responses and projected Hough centers. Hence, we can build a two layer graph for any image, where the nodes in the first layer $\mathbb{I}$ (image graph) are the underlying random variables X_i for all pixels in the image, and the second layer $\mathbb{H}$ (Hough graph) contains their transformed counterparts $\mathcal{H}(X_i)$ in the Hough vote map. Figure 7.2 illustrates this two-layer graph setup.

The first step is to optimally extract a seed-region from the corresponding Hough graph $\mathbb{H}$. Therefore, we propose a novel paradigm denoted as *Hough regions* which formulates the seed pixel extraction step itself as a segmentation problem. A *Hough region* is defined a connected, arbitrarily shaped subset of graph nodes $\mathbf{H_l} \in \mathbb{H}$, which are projections of the pixels belonging to the object instance into the Hough space. The idea and benefit of our paradigm is that in the physical world Hough center votes are imperfect. Even despite recent research to decrease the Hough vector impurity [79, 151], the Hough center is not a single pixel. This arises from various sources of error such as changes in global and local scales, aspect ratios, articulation, etc. Perfect Hough maxima are unlikely and there will always exist inconsistent centroid votes. However,

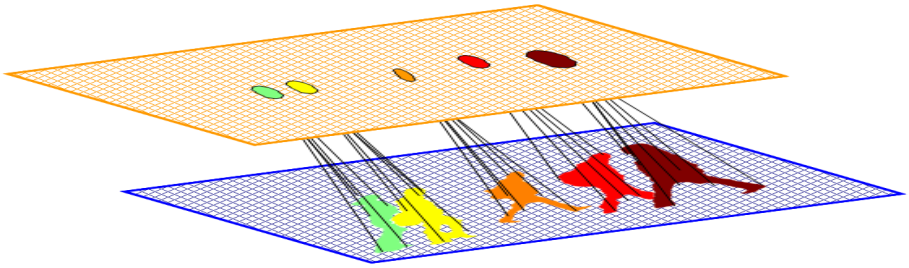


Figure 7.2: Two-layer graph design (top: Hough and bottom: image) to solve proposed image labeling problem. We identify *Hough regions* as subgraphs of the Hough graph $\mathbb{H}$. The back-projection into the image space, in combination with color information, distance transform values and contrast sensitive Potts potentials defines instance segmentations in the image graph $\mathbb{I}$.

Hough regions are designed exactly for this purpose: they capture the uncertainty not in a single Hough maximum point, but a region!

Thus, our goal is to identify *Hough regions* $\mathbf{H_l}$ in the Hough space $\mathbb{H}$, which then allows to directly define the corresponding seed region $\mathbf{S_l}$ by the back-projections into the image space, i. e. the random variables $X_i \in \mathbf{S_l}$ are the nodes in the image graph $\mathbb{I}$ which project into the *Hough region* $\mathbf{H_l}$. For this reason, we define a binary image labeling problem in the Hough graph by

$$E(x) = \sum_{X_i \in \mathcal{X}} \Theta(\mathcal{H}(X_i)) + \sum_{\mathcal{N}} \Phi(\mathcal{H}(X_i), \mathcal{H}(X_j)), \tag{7.8}$$

which contains the projected class-specific unary potentials Θ and a pairwise potentia Φ. The potential Θ for each variable X_i is projected to the Hough graph using $\mathcal{H}(X_i)$. The unary potential at any node $\mathcal{H}(X_i)$ is then the sum of the classifier responses voting for this node, as it is common in general Hough voting methods. The pairwise potential Φ is defined on this very same graph $\mathbb{H}$ as a gradient-sensitive Potts model, where the gradient is calculated as the difference between the unary potentials of two nodes $\mathcal{H}(X_i)$ and $\mathcal{H}(X_j)$ in the Hough graph $\mathbb{H}$. This binary labeling problem can be solved in a global optimal manner using any available graph cut based inference algorithm [31], and the obtained Hough region $\mathbf{H_l}$ is then back-projected to the seed region $\mathbf{S_l}$.

The second step, after finding the optimal seed region $\mathbf{S_l}$, is to identify all supporting pixels of the category instance in the image graph $\mathbb{I}$. Since we have now given the required seed region $\mathbf{S_l}$, we can apply our energy minimization problem defined in Equation 7.3 and again solve a binary labeling problem, for assigning each pixel to the

background or the currently analyzed category instance. The final part is the distance function $\Delta(X_i, \mathbf{S_l})$, which we define as

$$\Delta(X_i, \mathbf{S_l}) = \begin{cases} 0 & \text{if} \quad \mathcal{H}(X_i) \in \mathbf{H_l} \\ DT(X_i) & \text{otherwise} \end{cases}, \tag{7.9}$$

where $\mathcal{H}(X_i)$ is the Hough transformation of an image location to its associated Hough nodes and $DT(X_i)$ is the distance transform over the elements of the current seed pixels, which are given by the *Hough region* $\mathbf{H_l}$. Again this binary labeling problem can be solved using graph cut methods [31] and returns a binary segmentation mask $\mathcal{M}_l$ for the current category instance.

After finding the optimal segmentation $\mathcal{M}_l$ for the currently analyzed category instance in the image space, we update the Hough vote map, considering the already assigned Hough votes. In such a way our framework is not error-prone to spurious incorrect updates in the Hough voting space as it is common in related methods. This directly improves the detection performance, as occluded object instances are not removed. On the contrary, occluded instances are now more easily detectable by their visible segmentation, as they require less visibility with competing object instances or other occlusions.

In detail, the update considers each image location and its (independent) votes, which are accumulated in the nodes of the Hough graph. An efficient update is possibly by subtracting the previously segmented object instance from classwise potentials, which are initially $\Theta_0(X_i) = \Theta(X_i)$, by

$$\Theta_{t+1}(X_i) = \Theta_t(X_i) - \Theta(X_{M_l}), \tag{7.10}$$

where each random variable X_{M_l} within the segmentation mask $\mathcal{M}_l$ of the category instance is used to reduce the full graph. Using our obtained segmentations, we focus the update solely on the areas of the graph where the current object instance plays a role. This leads to a much finer dissection of the image and Hough graphs for the detections and their image support, as shown in Figure 7.2.

After updating the Hough graph $\mathbb{H}$ with the same update step, we repeat finding the next optimal seed region and afterwards segmenting the corresponding category instance in the image graph $\mathbb{I}$. This guarantees a monotonic decrease in $max(\Theta_{t+1})$ and our iteration stops when $max(\Theta_{t+1}) < p_{back}$, i. e. we have identified all Hough regions (object instances $\mathcal{L}$) above a threshold.

7.3.3 Comparison to Related Approaches

Our implicit step of updating the Hough vote maps by considering optimal segmentations in the image space, is related to other approaches in the field of non-maximum suppression. In general, one can distinguish two different approaches in this field.

Bounding boxes are frequently selected as underlying representation to perform non-maximum suppression and are the de-facto standard for defining the number of detections from a Hough image. In these methods the Hough space is analyzed using a Parzen-window estimate based on the Hough center votes. Local maxima in the Parzen window estimate determine candidates for the seed regions $\mathbf{S_l}$ by placing a bounding box considering the current scale onto the local maxima. Conflicting bounding boxes, e. g. within a certain distance and quality with respect to each other are removed (NMS). Each bounding box finally represents one seed region $\mathbf{S_l}$ and additionally defines a binary distance function $\Delta\left(X_i, \mathbf{S_l}\right)$ by setting $\Delta = 1$ for all pixels within the bounding box and $\Delta = 0$ to all other pixels.

Non-maximum suppression comes in many different forms like: a) Finding all local maxima and performing non-maximum suppression in terms of mutual bounding box overlap in image space to discard low scoring detections. b) Extracting global maxima iteratively and eliminating these by deleting the interior of the bounding box hypothesis in Hough space [79]. c) Iteratively finding global maxima for the seed pixels [14] and estimating bounding boxes and subtracting the corresponding Hough vectors. All pixels inside the bounding box are used to reduce the Hough vote map. This leads to a coarse dissection of the Hough map which leads to problems since it includes much background information and overlapping object instances.

Clustering methods allow more elegant formulation since they attempt to group Hough vote vectors to form the set of seed variables $\mathbf{S_l}$. The Implicit Shape Model (ISM) [121], for example, employs a mean-shift clustering on sparse Hough center locations to find coherent votes. They define the distance function $\Delta\left(X_i, \mathbf{S_l}\right)$ in terms of an L^2 norm on the Hough vectors $\mathcal{H}_i$. The work by Ommer and Malik [152] extends this to clustering Hough voting lines, which are infinite lines as opposed to finite Hough vectors. The benefit lies in extrapolating the scale from coherent Hough lines, as the line intersection in the 3D x–y–scale space determines the optimal scale. In general, these methods elegantly find the seed pixel selection by considering them as clusters,

however such approaches assumes well-distinguishable cluster distributions and are not well-suited for scenarios including many overlaps and occlusions. Furthermore, similar to the bounding box methods which require a Parzen-window estimate to bind connected Hough votes, clustering methods require an intra- to inter cluster balance. In terms of a mean-shift approach this is defined by the bandwidth parameter considered. An unfortunate drawback of clustering methods is often the limited scalability in terms of number of input Hough vectors that can be efficiently handled. An interesting alternative is the grouping of dependent object parts [214], which yields significantly less uncertain group votes compared to the individual votes.

Our proposed approach has several advantages compared to the discussed related methods. A key benefit is, that we do not have to fix a range for local neighborhood suppression, as it is for example required in non-maximum suppression approaches. As it is also demonstrated in the experimental section, our method is robust to an increased range of variations in object articulations, aspect ratios and scales, which for example allows to reduce the number of analyzed scales during testing. We implicitly also provide segmentations of all detected category instances, without requiring segmentation masks during training. In overall, the benefits of our *Hough Regions* approach lead to higher precision and recall compared to related approaches.

7.4 Experimental Evaluation

In general, we can use any available general Hough voting method as starting point for our method. The most related approaches for non-maxima suppression are [14, 79]. We use their publicly available code to provide required class likelihoods and the centroid votes. For not overlapping datasets like PASCAL VOC, we achieve the same performance as our baseline method. For better direct comparison, we therefore evaluate against their challenging datasets for overlapping detections, namely TUD crossing and TUD campus. Additionally, we evaluate on a novel window detection dataset GT240 designed for testing aspect ratio and distortion. We use two GMM components, equal weighting between the energy terms and a background constant $p_{back} = 0.125$. The overall runtime of our method is a moderate 10 seconds on average per image.

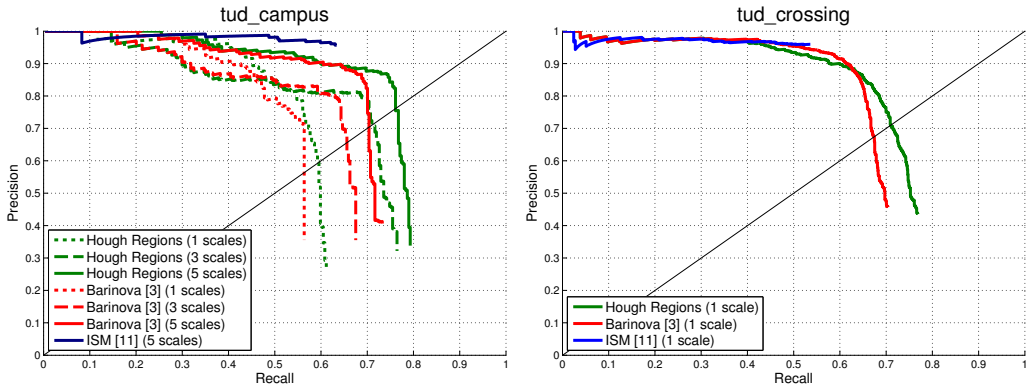


Figure 7.3: Object detection performance as Recall/Precision curve for the TUD campus and TUD crossing sequences showing analysis at multiple scales for highly overlapping instances. See text for details.

7.4.1 TUD Campus

To demonstrate the ability of our system to deal with occluded and overlapping object instances (which results in an increased recall using the PASCAL 50% criterion), we evaluated on the TUD campus sequence, which contains 71 images and 303 highly overlapping pedestrians with large scale changes. We evaluated the methods on multiple scale ranges (one, three and five scales) and Figure 7.3 shows the Recall-Precision curve (RPC) for the TUD campus sequence. Aside the fact that multiple scales benefit the detection performance for all methods, one can also see that our *Hough regions* method surpasses the performance at each scale range (+3%, +8%, +8% over [14]) and over fewer scales. For example, using *Hough regions* we only require three scales to achieve the recall and precision, which is otherwise only achieved using five scales. Hence, our methods is better in handling scale changes, which requires fewer scales during test time and thus reduced runtime complexity. The ISM approach of Leibe et al. [121] reaches a very good precision, but at the costs of recall. The MDL criterion limits their ability to find all partially occluded objects, resulting in a lower recall.

7.4.2 TUD Crossing

The TUD crossing dataset [6] contains 201 images showing profile views of pedestrians in a relatively crowded scenario. The annotation by Andriluka et al. [6] contains 1008 tight bounding boxes designed for pedestrians with at least 50% visibility, ignoring highly overlapping pedestrians. For this reason we created a segmentation-accurate

annotation. The new annotation is based on the original bounding box annotation and now contains 1212 bounding box annotations with corresponding segmentations. In addition to bounding boxes we also annotated the visibility of pedestrians in fully visible body parts: head, front part of upper body, back part of upper body, left leg or right leg.

Typically, in this sequence three scales are evaluated, however to show ability to handle scale, we evaluate only on a single scale for all methods. Figure 7.3 shows the Recall-Precision curve (RPC) for the TUD crossing sequence in comparison to an Implicit Shape Model (ISM) [121] and the probabilistic framework of [14]. Our method achieves a better recall compared to the other approaches. We increase the recall as well as precision indicating that our method can better handle the overlaps, scale and articulation changes.

7.4.3 GT240

We also evaluated our method on a novel street-level window detection dataset (denoted as GT240), which consists of 240 buildings with 5400 redundant images with a total of 5542 window instances. Window detection itself is difficult due to immense intra-class appearance variations. Additionally, the dataset includes a large range of diverging aspect ratios and strong perspective distortions, which usually limit object detectors. In our experiment we trained a standard detection framework based on [79] and tested on three different scales and a single aspect ratio. As shown in Figure 7.4 we can substantially improve the detection performance in both recall and precision (12% at EER) compared to the baseline. Our *Hough regions* based detector consistently delivers improved localization performance better suppressing false positives and handling different aspect ratios and perspective distortions, because we are not limited to axis-aligned bounding boxes.

7.4.4 Segmentation

As final experiment we analyze achievable segmentation accuracy on the TUD crossing sequence [6], where we created binary segmentations masks for each object instance for the entire sequence. Segmentation performance is measured by the ratio between the number of correctly assigned pixels to the the total number of pixels (segmentation accuracy). We compared our method to the Implicit Shape Model (ISM) [121], which also provides segmentations for each detected instance. Please note, that the ISM requires segmentation masks for all training samples to provide reasonable segmentations

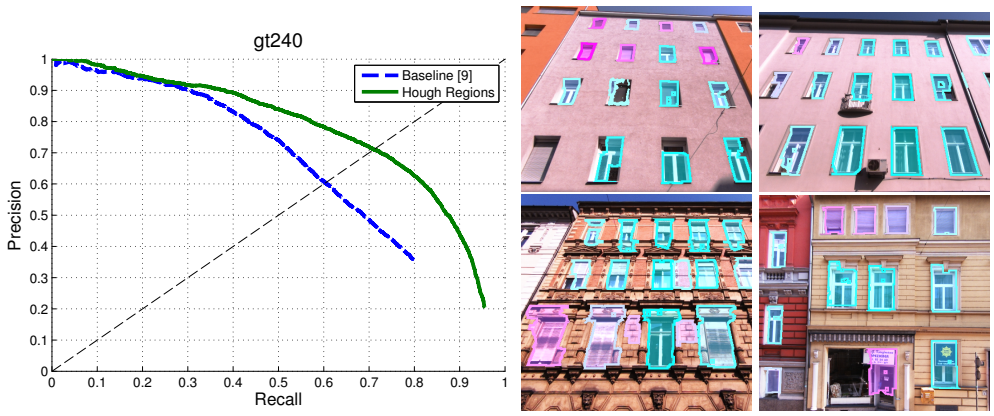


Figure 7.4: Object detection performance as Recall/Precision curve on the window detection dataset GT240 with strong distortion and aspect ratio changes in 5400 images.

Figure 7.5: Illustration of the segmentations for TUD crossing and TUD campus datasets.

whereas our method learns from training images only. Nevertheless, we achieve competitive segmentation accuracy of 98.59% compared to 97.99% for the ISM. Illustrative results are shown in Figure 7.5.

7.5 Conclusion

In this chapter we proposed an object detection method to jointly solve instance localization and segmentation. It requires only a generic classifier response, which gives the feature-wise class probabilities and the object center location. Our novel *Hough regions* determine the locations of object instances, which implicitly handles large location uncertainty due to articulation, aspect ratio and scale. As shown in the experiments, our method jointly and accurately delineates the location and outline of object instances, leading to increased detection recall and precision for overlapping detections. These results confirm related research where combining the tasks of detection and segmentation improves performance, because of the joint optimization benefit over separate individual solutions. Future work entails using our method during the training of classifiers to provide more accurate estimation of foreground and background without increase of manual supervision. Further, an analysis whether verification of the segmentation rejects false positives and improves the precision is interesting.

I never see what has been done;
I only see what remains to be done.

Buddha

8

Conclusion and Outlook

In this thesis we addressed the following topics in computer vision: mid-level information for contour extraction, structured description of shape, partial matching of this structured shape, grouping of partial contour matches, discriminative learning of shape category models, and joint localization and segmentation of multiple object instances. The main underlying feature in the proposed solutions is shape, as it is invariant to changes in texture and illumination yet still captures the common properties of objects from the same category.

The contributions of the thesis include a novel integration of mid-level context cues from regions to increase the stability and precision of contour extraction. A novel structured description of shape, denoted as *Structural Measurement Descriptor* (SMD), exploiting the sequence of points to include hierarchical and partial shape information, which can be efficiently accessed and compared. The structural descriptor is generic, which allows various features and different levels of shape abstraction, for example closed regions, open contours or local contour fragments. A novel partial matching method is proposed for the structured description independent of the features, which results in an efficient 3D tensor similarity for arbitrary length matches. This is employed for partial region matching for shape retrieval. A novel partial contour aggregation, which

summarizes redundant partial matches for performing object detection in cluttered images, is proposed. Further, a novel method for joint shape fragment and category model learning for object detection is shown. And finally, a novel formulation for joining classification, localization and segmentation is described using a two-layer image and Hough graph for enumeration of object instances to increase the robustness over overlapping instances and changes in scale and aspect ratio.

Future work has many directions and in the following a few topics are discussed. The overall goal is a method which entails all parts joining the shape and appearance feature extraction up to the final segmentation and scene explanation. Additionally the range of applications for shape-based description and matching as it has been introduced here may be extended to partial shape retrieval in user drawings, tracking of contours in time or space for multiple view reconstruction.

Features are the essential part of contour extraction and shape description. Future work will investigate the integration of the proposed mid-level context cues from region into related contour detection and semantic classification approaches. Further research may also be directed to evaluating the similarity of contour during their extraction. In terms of the component tree analysis the individual similarity of regions can directly be computed by comparing the region to the category model. This will result in an category-specific contour extraction and further to an object instance localization.

Description of features requires a notion of support. The design of support windows for local shape fragment description will result in a higher discriminability of the descriptor. Similar to the support window from local interest points given a scale estimate, the support window may be estimated from the local curvature scale space. This will result in more suitable context for local fragment description and result in an increase of classification accuracy.

Non-rigid deformation of objects is a natural phenomenon, which occurs whenever the object undergoes changes in pose and articulation. In this work we have considered rigid template models and a partial similarity measure to determine matching parts of objects. An interesting direction for future work is to automatically determine rigid and non-rigid parts of an object and learn a weighted deformation model for these parts to incorporate into the partial matching cost.

Complexity is a vital topic in terms of the computational requirements. There is an additional speed up when moving from partial matching of contours to classifying overlapping local shape fragments due to the reduced complexity in description and matching. However, a loss in discriminative power is consequently present. Thus future work may focus on combining the partial matching and classification learning techniques. One direction is the introduction of the partial distance functions for hierarchical divide and conquer schemes.

Prior knowledge for specific category models is often a benefit for highly accurate results. In this work we placed a focus on generic object detection, and future work may focus on incorporating prior knowledge for specific object classes. For example, pedestrian detection and segmentation will benefit from an estimation of the human pose, shape priors and reasoning about visibility of parts.

Grouping is one of the most vital parts of retrieving partial matches. The current methods aim for either maximizing the length of matches for regions or extracting multiple highly similar partial matches of image contour. A future direction for the matching is to incorporate regularization terms, which optimize over length, similarity and fragmentation. The additional energies in the optimization over the best partial matches will help group the similarity and partiality to more coherent parts. One idea, for example, is to incorporate a constraint on the number of parts and not only the similarity and length of matches.

A second direction entails joining high-level information into the grouping optimization for part matching. Future research will look into holistic grouping as a role in selection of valid partial matches. This will discard many ghost matches due to unlikely inclusion in accordance with the global holistic layout of the category model.

Joint Understanding is the overall final goal, where one jointly estimates the class of the object (classification), the position (localization) and the spatial extent (segmentation). This direction entails not only modeling these three aspects, but further looking at the relationships between multiple object instances (interaction) and the underlying scene structure (surface geometry). Future research will investigate novel optimization strategies to update each task with knowledge from the other tasks and provide a holistic interrelated joint understanding.

Careful. We don't want to learn from this.

Bill Watterson, "Calvin and Hobbes"

A Publications

- Hayko Riemenschneider, Ulrich Krispel, Wolfgang Thaller, Michael Donoser, Sven Havemann, Dieter Fellner, and Horst Bischof. Irregular lattices for complex shape grammar facade parsing. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012.

- Hayko Riemenschneider, Michael Donoser, and Horst Bischof. Image retrieval by shape-focused sketching of objects. In Proceedings of the Computer Vision Winter Workshop (CVWW), 2011.

- Peter Kontschieder, Hayko Riemenschneider, Michael Donoser, and Horst Bischof. Discriminative Learning of Contour Fragments for Object Detection. In Proceedings of the British Machine Vision Conference (BMVC), 2011.

- Michael Donoser, Martin Urschler, Hayko Riemenschneider, and Horst Bischof. Highly Consistent Sequential Segmentation. In Proceedings of Scandinavian Conference on Image Analysis (SCIA), 2011.

- Hayko Riemenschneider, Michael Donoser, and Horst Bischof. Using Partial Edge Contour Matches for Efficient Object Category Localization. In Proceedings of European Conference on Computer Vision (ECCV), 2010.

- Michael Donoser, Hayko Riemenschneider, and Horst Bischof. Linked Edges as Stable Region Boundaries. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010.
- Michael Donoser, Hayko Riemenschneider, and Horst Bischof. IS-Match: Partial Shape Matching by Efficiently Solving an Order Preserving Assignment Problem. In IPSJ Transactions on Computer Vision and Applications, 2010.
- Michael Donoser, Hayko Riemenschneider, and Horst Bischof. Shape Prototype Signatures for Action Recognition. In Proceedings of International Conference on Pattern Recognition (ICPR), 2010.
- Michael Donoser, Hayko Riemenschneider, and Horst Bischof. Shape Guided Maximally Stable Extremal Region (MSER) Tracking. In Proceedings of International Conference on Pattern Recognition (ICPR), 2010.
- Sabine Sternig, Hayko Riemenschneider, Peter M. Roth, Michael Donoser, and Horst Bischof. Robust Person Detection by Classifier Cubes and Local Verification. In Workshop of the Austrian Association for Pattern Recognition (OAGM), 2010.
- Michael Donoser, Hayko Riemenschneider, and Horst Bischof. Efficient Partial Shape Matching of Outer Contours. In Proceedings of Asian Conference on Computer Vision (ACCV), 2009.
- Hayko Riemenschneider, Michael Donoser, and Horst Bischof. Bag of Optical Flow Volumes for Image Sequence Recognition. In Proceedings of the British Machine Vision Conference (BMVC), 2009.
- Hayko Riemenschneider, Michael Donoser, and Horst Bischof. Finding Stable Extremal Region Boundaries. In Workshop of the Austrian Association for Pattern Recognition (OAGM), 2009.
- Hayko Riemenschneider, Michael Donoser, and Horst Bischof. Robust Online Object Learning and Recognition by MSER Tracking. In Proceedings of the Computer Vision Winter Workshop (CVWW), 2008.
- Hayko Riemenschneider, Michael Donoser, and Horst Bischof. Online Object Recognition by MSER Trajectories. In Proceedings of International Conference on Pattern Recognition (ICPR), 2008.

If I have seen further it is by standing on the shoulders of giants.

Isaac Newton

B Bibliography

[1] Ackermann, W. (1928). Zum Hilbertschen Aufbau der reelen Zahlen. *Mathematische Annalen*, 99:118–133.

[2] Aichholzer, O., Aurenhammer, F., and Gärtner, B. (1995). A novel type of skeleton for polygons. *Journal of Universal Computer Science*.

[3] Alt, H. and Godau, M. (1995). Computing the fréchet distance between two polygonal curves. *International Journal of Computational Geometry and Applications (IJCGA)*.

[4] Amit, Y., August, G., and Geman, D. (1996). Shape quantization and recognition with randomized trees. *Neural Computation*.

[5] Amit, Y., Geman, D., and Fan, X. (2004). A coarse-to-fine strategy for multiclass shape detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[6] Andriluka, M., Roth, S., and Schiele, B. (2008). People-tracking-by-detection and people-detection-by-tracking. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[7] Arbelaez, P. (2006). Boundary extraction in natural images using ultrametric contour maps. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[8] Arbelaez, P., Maire, M., Fowlkes, C., and Malik, J. (2009). From contours to regions. an empirical evaluation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[9] Arbelaez, P., Maire, M., Fowlkes, C., and Malik, J. (2011). Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[10] Arica, N. and Vural, F. Y. (2003). BAS: a perceptual shape descriptor based on the beam angle statistics. *Pattern Recognition Letters*.

[11] Arkin, E., Chew, L., Huttenlocher, D., Kedem, K., and Mitchell, J. (1991). Determining the similarity of deformable shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[12] Bai, X., Li, Q., Latecki, L., Liu, W., and Tu, Z. (2009). Shape band: A deformable object detection approach. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[13] Ballard, D. (1981). Generalizing the hough transform to detect arbitrary shapes. *Pattern Recognition (PR)*.

[14] Barinova, O., Lempitsky, V., and Kohli, P. (2010a). On detection of multiple object instances using hough transform. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[15] Barinova, O., Lempitsky, V., and Kohli, P. (2010b). On the detection of multiple object instances using hough transforms. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[16] Barrow, H., Tenenbaum, J., Bolles, R., and Wolf, H. (1977). Parametric correspondence and chamfer matching: Two new techniques for image matching. In *International Joint Conference on Artificial Intelligence (IJCAI)*.

[17] Basri, R., Costa, L., Geiger, D., and Jacobs, D. (1998). Determining the similarity of deformable shapes. *Vision Research*.

[18] Belongie, S., Malik, J., and Puzicha, J. (2000). Shape Context: A new descriptor for shape matching and object recognition. In *Advances in Neural Information Processing Systems (NIPS)*.

[19] Belongie, S., Malik, J., and Puzicha, J. (2002). Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[20] Berg, A., Berg, T., and Malik, J. (2005). Shape matching and object recognition using low distortion correspondences. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[21] Biederman, I. (1985). Human image understanding: Recent research and a theory. In *Computer Vision, Graphics, and Image Processing*, volume 32, pages 29–73.

[22] Biederman, I. (1987). Recognition-by-components: a theory of human image understanding. *Psychological Review*, 94:115–147.

[23] Biederman, I. and Ju, G. (1988). Surface vs. edge-based determinants of visual recognition. *Cognitive Psychology*.

[24] Biswas, S., Aggarwal, G., and Chellappa, R. (2007). Efficient indexing for articulation invariant shape matching and retrieval. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[25] Blum, H. (1967). A transformation for extracting new descriptors of shape. *Models for the perception of speech and visual form*.

[26] Bookstein, F. (1986). Size and Shape Spaces for Landmark Data in Two Dimensions. *Statistical Science*.

[27] Bookstein, F. (1991). *Morphometric tools for landmark data*. Cambridge University Press.

[28] Borenstein, E. and Ullman, S. (2002). Class-specific, top-down segmentation. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[29] Borenstein, E. and Ullman, S. (2004). Learning to segment. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[30] Borgefors, G. (1988). Hierarchical chamfer matching: A parametric edge matching algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[31] Boykov, Y., Veksler, O., and Zabih, R. (2001). Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[32] Breiman, L. (2001). Random forests. *Machine Learning*.

[33] Brendel, W. and Todorovic, S. (2009). Video object segmentation by tracking regions. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[34] Bronstein, A., Bronstein, M., Bruckstein, A., and Kimmel, R. (2008). Partial similarity of objects, or how to compare a centaur to a horse. *International Journal of Computer Vision (IJCV)*.

[35] Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[36] Catanzaro, B., Su, B.-Y., Sundaram, N., Lee, Y., Murphy, M., and Keutzer, K. (2009). Efficient, high-quality image contour detection. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[37] Chen, L., Feris, R., and Turk, M. (2008). Efficient partial shape matching using smith-waterman algorithm. In *Proceedings of Workshop on Non-Rigid Shape Analysis and Deformable Image Alignment (NORDIA)*.

[38] Chui, H. and Rangarajan, A. (2003). A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding (CVIU)*.

[39] Comaniciu, D. and Meer, P. (2002). Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[40] Cootes, T., Cooper, D., Taylor, C., and Graham, J. (1992). A trainable method of parametric shape description. *Image and Vision Computing*.

[41] Cootes, T. and Taylor, C. (1992). Active shape models. In *Proceedings of British Machine Vision Conference (BMVC)*.

[42] Couprie, M., Najman, L., and Bertrand, G. (2005). Quasi-linear algorithms for the topological watershed. *Journal of Mathematical Imaging and Vision (JMIV)*.

[43] Crandall, D., Felzenszwalb, P., and Huttenlocher, D. (2005). Spatial priors for part-based recognition using statistical models. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[44] Crow, F. (1984). Summed-area tables for texture mapping. In *Proceedings of Conference on Computer Graphics and Interactive Techniques (CCGIT)*.

[45] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[46] Desai, C. and Ramanan, D. (2009). Discriminative models for multi-class object layout. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[47] Desai, C., Ramanan, D., and Fowlkes, C. (2011). Discriminative models for multi-class object layout. *International Journal of Computer Vision (IJCV)*.

[48] Dollar, P., Tu, Z., and Belongie, S. (2006). Supervised learning of edges and object boundaries. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[49] Donoser, M., Riemenschneider, H., and Bischof, H. (2009). Efficient Partial Shape Matching of Outer Contours. In *Proceedings of Asian Conference on Computer Vision (ACCV)*.

[50] Donoser, M., Riemenschneider, H., and Bischof, H. (2010a). IS-Match: Partial Shape Matching by Efficiently Solving an Order Preserving Assignment Problem. In *IPSJ Transactions on Computer Vision and Applications*.

[51] Donoser, M., Riemenschneider, H., and Bischof, H. (2010b). Linked Edges as Stable Region Boundaries. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[52] Donoser, M., Riemenschneider, H., and Bischof, H. (2010c). Shape Guided Maximally Stable Extremal Region (MSER) Tracking. In *Proceedings of International Conference on Pattern Recognition (ICPR)*.

[53] Donoser, M., Urschler, M., Riemenschneider, H., and Bischof, H. (2011). Highly Consistent Sequential Segmentation. In *Proceedings of Scandinavian Conference on Image Analysis (SCIA)*.

[54] Duchon, J. (1976). Splines minimizing rotation invariant semi-norms in Sobolev spaces. *Constructive Theory of Functions of Several Variables*.

[55] Duda, R. and Hart, P. (1972). Use of the hough transformation to detect lines and curves in pictures. *Comminications of the ACM*.

[56] Efrat, A. and Itai, A. (1996). Improvements on bottleneck matching and related problems using geometry. In *Proceedings of International Symposium on Computational Geometry (ISCG)*.

[57] Felzenszwalb, P. (2003). Representation and detection of deformable shapes. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[58] Felzenszwalb, P. (2005). Representation and detection of deformable shapes. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[59] Felzenszwalb, P., Girshick, R., McAllester, D., and Ramanan, D. (2010). Object detection with discriminatively trained part based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[60] Felzenszwalb, P. and Huttenlocher, D. (2005). Pictorial structures for object recognition. *International Journal of Computer Vision (IJCV)*.

[61] Felzenszwalb, P. and Schwartz, J. (2007). Hierarchical matching of deformable shapes. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[62] Fergus, R., Perona, P., and Zisserman, A. (2003). Object class recognition by unsupervised scale-invariant learning. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[63] Ferrari, V., Fevrier, L., Jurie, F., and Schmid, C. (2008). Groups of adjacent contour segments for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[64] Ferrari, V., Jurie, F., and Schmid, C. (2007). Accurate object detections with deformable shape models learnt from images. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[65] Ferrari, V., Jurie, F., and Schmid, C. (2009). From images to shape models for object detection. *International Journal of Computer Vision (IJCV)*.

[66] Ferrari, V., Tuytelaars, T., and van Gool, L. (2006). Object detection by contour segment networks. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[67] Fidler, S., Berginc, G., and Leonardis, A. (2006). Hierarchical statistical learning of generic parts of object structure. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[68] Fidler, S., Boben, M., and Leonardis, A. (2008). Similarity-based cross-layered hierarchical representation for object categorization. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[69] Fidler, S., Boben, M., and Leonardis, A. (2009a). *Object Categorization: Computer and Human Vision Perspectives*, chapter Learning Hierarchical Compositional Representations of Object Structure. Cambridge University Press.

[70] Fidler, S., Boben, M., and Leonardis, A. (2009b). Optimization framework for learning a hierarchical shape vocabulary for object class detection. In *Proceedings of British Machine Vision Conference (BMVC)*.

[71] Fidler, S., Boben, M., and Leonardis, A. (2010). A coarse-to-fine taxonomy of constellations for fast multi-class object detection. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[72] Fidler, S. and Leonardis, A. (2007). Towards scalable representation of object categories: Learning a hierarchy of parts. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[73] Fischler, M. and Elschlager, R. (1973). The representation and matching of pictorial structures. *IEEE Transactions on Computer*.

[74] Floros, G., Rematas, K., and Leibe, B. (2011). Multi-Class Image Labeling with Top-Down Segmentation and Generalized Robust P^N Potentials. In *Proceedings of British Machine Vision Conference (BMVC)*.

[75] Fowlkes, C., Martin, D., and Malik, J. (2007). Local figure-ground cues are valid for natural images. *Journal of Vision*.

[76] Freeman, H. (1961). On the encoding of arbitrary geometric configurations. *IRE Transactions on Electronic Computers*.

[77] Freund, Y. and Schapire, R. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences (JCSS)*.

[78] Frey, B. and Dueck, D. (2007). Clustering by passing messages between data points. *Science*, 315:972–976.

[79] Gall, J. and Lempitsky, V. (2009). Class-specific hough forests for object detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[80] Gavrila, D. (2000). Pedestrian detection from a moving vehicle. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[81] Gavrila, D. (2007). A bayesian, exemplar-based approach to hierarchical shape matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[82] Gdalyahu, Y. and Weinshall, D. (1999). Flexible syntactic matching of curves and its application to automatic hierarchical classification of silhouettes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[83] Geman, S. and Geman, D. (1984). Stochastic relaxation, gibbs distributions and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[84] Goodall, C. (1991). Procrustes methods in the statistical analysis of shape. *Journal of the Royal Statistical Society*.

[85] Gorelick, L., Galun, M., and Brandt, A. (1996). Shape representation and classification using the poisson equation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[86] Gould, S., Gao, T., and Koller, D. (2009). Region-based segmentation and object detection. In *Advances in Neural Information Processing Systems (NIPS)*.

[87] Govindu, V. (2005). Tensor decomposition for geometric grouping and segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[88] Grauman, K. and Darrell, T. (2004). Fast contour matching using approximate earth mover's distance. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[89] Grauman, K. and Darrell, T. (2005). The pyramid match kernel: Discriminative classification with sets of image features. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[90] Gu, C., Lim, J., Arbelaez, P., and Malik, J. (2009). Recognition using regions. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[91] Haar, A. (1910). Zur Theorie der orthogonalen Funktionensysteme. *Mathematische Annalen*, 69:331–371.

[92] Hartigan, J. (1985). Statistical theory in clustering. *Journal of Classification (JOC)*.

[93] Hofmann, T. and Buhmann, J. (1997). Pairwise data clustering by deterministic annealing. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[94] Hoiem, D., Stein, A., Efros, A., and Hebert, M. (2007). Recivering occlusion boundaries from a single image. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[95] Hough, P. (1959). Machine analysis of bubble chamber pictures. In *International Conference on High Energy Accelerators and Instrumentation*.

[96] Hu, M. (1962). Visual Pattern Recognition by Moment Invariants. *IRE Transactions on Information Theory*, 8:179–187.

[97] Huttenlocher, D., Klanderman, G., and Rucklidge, W. (1993). Comparing images using the Hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[98] Jeannin, S. and Bober, M. (1999). Description of core experiments for MPEG-7 motion/shape. Technical report, MPEG-7, ISO/IEC JTC1/SC29/WG11/MPEG99/N2690.

[99] Jones, R. (1997). Component trees for image filtering and segmentation. In *IEEE Workshop on Nonlinear Signal and Image Processing (NSIP)*.

[100] Jones, R. (1999). Connected filtering and segmentation using component trees. *Computer Vision and Image Understanding (CVIU)*.

[101] Jurie, F. and Schmid, C. (2004). Scale-invariant shape features for recognition of object categories. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[102] Kendall, D. (1984). Shape Manifolds, Procrustean Metrics, and Complex Projective Spaces. *Bulletin of the London Mathematical Society*.

[103] Khotanzad, A. and Hong, Y. (1990). Invariant image recognition by Zernike moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[104] Koffka, K. (1935). *Principles of Gestalt Psychology*. Lund Humphries.

[105] Kohli, P., Ladicky, L., and Torr, P. (2009). Robust higher order potentials for enforcing label consistency. *International Journal of Computer Vision (IJCV)*.

[106] Kokkinos, I. and Yuille, A. (2009). Hop: Hierarchical object parsing. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[107] Konishi, S., Yuille, A., Coughlan, J., and Zhu, S. (2003). Statistical edge detection: learning and evaluating edge cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[108] Kontschieder, P., Riemenschneider, H., Donoser, M., and Bischof, H. (2011). Discriminative Learning of Contour Fragments for Object Detection. In *Proceedings of British Machine Vision Conference (BMVC)*.

[109] Kovesi, P. (2008). MATLAB and Octave functions for computer vision and image processing. Centre for Exploration Targeting, School of Earth and Environment, The University of Western Australia. <http://www.csse.uwa.edu.au/∼pk/research/matlabfns/>.

[110] Kuhn, H. (1955). The Hungarian Method for the assignment problem. In *Naval Research Logistics Quarterly*, volume 2, pages 83–97.

[111] Kuijper, A. and Olsen, O. (2006). Describing and matching 2D shapes by their points of mutual symmetry. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[112] Ladicky, L. (2011). *Global Structured Models towards Scene Understanding*. Phd thesis, University of Oxford.

[113] Ladicky, L., Russell, C., Kohli, P., and Torr, P. (2009). Associative Hierarchical CRFs for Object Class Image Segmentation. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[114] Ladicky, L., Sturgess, P., Alahari, K., Russell, C., and Torr, P. (2010). What, Where & How Many? Combining Object Detectors and CRFs. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[115] Larlus, D. and Jurie, F. (2008). Combining appearance models and markov random fields for category level object segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[116] Latecki, L., Lakaemper, R., and Wolter, D. (2005). Optimal partial shape similarity. *Image and Vision Computing Journal (IVCJ)*.

[117] Latecki, L., Lakämper, R., and Eckhardt, U. (2000). Shape descriptors for non-rigid shapes with a single closed contour. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[118] Latecki, L., Megalooikonomou, V., Wang, Q., and Yu, D. (2007). An elastic partial shape matching technique sequences. *Pattern Recognition (PR)*.

[119] Lazebnik, S., Schmid, C., and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[120] Leibe, B., Leonardis, A., and Schiele, B. (2004). Combined object categorization and segmentation with an implicit shape model. In *Workshop on Statistical Learning in Computer Vision*.

[121] Leibe, B., Leonardis, A., and Schiele, B. (2008). Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision (IJCV) - Special Issue on Learning for Recognition and Recognition for Learning*.

[122] Leordeanu, M., Hebert, M., and Sukthankar, R. (2007). Beyond local appearance: Category recognition from pairwise interactions of simple features. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[123] Li, F., Carreira, J., and Sminchisescu, C. (2010). Object recognition as ranking holistic figure-ground hypotheses. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[124] Lin, C. and Chellappa, R. (1987). Classification of partial 2-d shapes using fourier descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[125] Lin, Z., Davis, L., Doermann, D., and DeMenthon, D. (2007). Hierarchical part-template matching for human detection and segmentation. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[126] Ling, H. and Jacobs, D. (2005). Using the Inner-Distance for Classification of Articulated Shapes. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[127] Ling, H. and Jacobs, D. (2007). Shape classification using the inner-distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[128] Ling, H. and Okada, K. (2007). An efficient earth mover's distance algorithm for robust histogram comparison. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[129] Liu, M., Tuzel, O., Veeraraghavan, A., and Chellappa, R. (2010). Fast directional chamfer matching. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[130] Lowe, D. (1984). *Perceptual Organization and Visual Recognition*. Phd thesis, Stanford University.

[131] Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*.

[132] Lu, C., Latecki, L., Adluru, N., Ling, H., and Yang, X. (2009). Shape guided contour fragment grouping with particle filters. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[133] Ma, T. and Latecki, L. (2011). From partial shape matching through local deformation to robust global shape similarity for object detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[134] Maji, S. and Malik, J. (2009). Object detection using a max-margin hough transform. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[135] Marr, D. (1982). *Vision. A computational investigation into the human representation and processing of visual information*. W. H. Freeman.

[136] Martin, D., Arbelaez, P., Fowlkes, C., and Malik, J. (2008). Using contours to detect and localize junctions in natural images. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[137] Martin, D., Fowlkes, C., and Malik, J. (2004). Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[138] Martin, D., Fowlkes, C., Tal, D., and Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[139] Matas, J., Chum, O., Urban, M., and Pajdla, T. (2002). Robust Wide Baseline Stereo From Maximally Stable Extremal Regions. In *Proceedings of British Machine Vision Conference (BMVC)*.

[140] Mian, A., Bennamoun, M., and Owens, R. (2006). Three-dimensional model-based object recognition and segmentation in cluttered scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[141] Mikolajczyk, K. and Schmid, C. (2005). A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[142] Minkowski, H. (1953). *Geometrie der Zahlen*. Chelsea.

[143] Mokhtarian, F., Abbasi, S., and Kittler, J. (1996). Efficient and robust retrieval by shape content through curvature scale space. In *Proceedings of International Workshop on Image Databases and Multimedia Search*.

[144] Mokhtarian, F. and Mackworth, A. (1992). A theory of Multiscale, Curvature-based Shape Representation for Planar Curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[145] Moosmann, F., Triggs, B., and Jurie, F. (2006). Fast discriminative visual codebooks using randomized clustering forests. In *Advances in Neural Information Processing Systems (NIPS)*.

[146] Mori, G., Belongie, S., and Malik, H. (2001). Shape contexts enable efficient retrieval of similar shapes. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[147] Mosorov, V. and Kowalski, T. M. (2002). The development of component tree structure for grayscale image segmentation. In *Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET)*.

[148] Najman, L. and Couprie, M. (2004). Quasi-linear algorithm for the component tree. In *SPIE Vision Geometry XII*.

[149] Najman, L. and Schmitt, M. (1996). Geodesic saliency of watershed contours and hierarchical segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[150] Nistér, D. and Stewénius, H. (2008). Linear time maximally stable extremal regions. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[151] Okada, R. (2009). Discriminative generalized hough transform for object dectection. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[152] Ommer, B. and Malik, J. (2009). Multi-scale object detection by clustering lines. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[153] Opelt, A., Pinz, A., and Zisserman, A. (2006a). A boundary-fragment-model for object detection. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[154] Opelt, A., Pinz, A., and Zisserman, A. (2006b). Incremental learning of object detectors using a visual shape alphabet. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[155] Osada, R., Funkhouser, T., Chazelle, B., and Dobkin, D. (2002). Shape distributions. *ACM Transactions on Graphics*.

[156] Palmer, S. (1999). *Vision science: Photons to phenomenology*. The MIT Press.

[157] Pavan, M. and Pelillo, M. (2003). Dominant sets and hierarchical clustering. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[158] Pavan, M. and Pelillo, M. (2007). Dominant sets and pairwise clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[159] Payet, N. and Todorovic, S. (2010). From a set of shapes to object discovery. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[160] Quinlan, R. (1986). Induction of decision trees. *Machine Learning*, 1:81–106.

[161] Ramanan, D. (2007). Using segmentation to verify object hypotheses. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[162] Ravishankar, S., Jain, A., and Mittal, A. (2008). Multi-stage contour based detection of deformable objects. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[163] Riemenschneider, H., Donoser, M., and Bischof, H. (2009). Finding Stable Extremal Region Boundaries. In *Proceedings of Austrian Association for Pattern Recognition (AAPR)*.

[164] Riemenschneider, H., Donoser, M., and Bischof, H. (2010). Using Partial Edge Contour Matches for Efficient Object Category Localization. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[165] Riemenschneider, H., Donoser, M., and Bischof, H. (2011). Image retrieval by shape-focused sketching of objects. In *Proceedings of Computer Vision Winter Workshop (CVWW)*.

[166] Riemenschneider, H., Sternig, S., Donoser, M., Roth, P., and Bischof, H. (Submitted). Hough Regions for Joining Instance Localization and Segmentation. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[167] Rubner, Y., Tomasi, C., and Guibas, L. (1998). A metric for distributions with applications to image databases. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[168] Saber, E., Xu, Y., and Tekalp, A. (2005). Partial shape recognition by sub-matrix matching for partial matching guided image labeling. *Pattern Recognition (PR)*.

[169] Salembier, P., Oliveras, A., and Garrido, L. (1998). Anti-extensive connected operators for image and sequence processing. *IEEE Transactions on Image Processing (TIP)*.

[170] Salukwadze, M. (1979). *Vector-Valued Optimization Problems in Control Theory*. Academic Press.

[171] Schapire, R. (1990). The Strength of Weak Learnability. *Machine Learning*, 5:197–227.

[172] Schindler, K. and Suter, D. (2008). Object detection by global contour shape. *Pattern Recognition (PR)*.

[173] Schmidt, F., Farin, D., and Cremers, D. (2007). Fast matching of planar shapes in sub-cubic runtime. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[174] Schroff, F., Criminisi, A., and Zisserman, A. (2008). Object class segmentation using random forests. In *Proceedings of British Machine Vision Conference (BMVC)*.

[175] Scott, C. and Nowak, R. (2006). Robust contour matching via the order-preserving assignment problem. *IEEE Transactions on Image Processing (TIP)*.

[176] Sebastian, T., Klein, P., and Kimia, B. (2004). Recognition of shapes by editing their shock graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[177] Sebastian, T. B., Klein, P. N., and Kimia, B. B. (2001). Recognition of shapes by editing shock graphs. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[178] Sebastian, T. B., Klein, P. N., and Kimia, B. B. (2003). On aligning curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[179] Selinger, A. and Nelson, R. (1998). A cubist approach to object recognition. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[180] Sharvit, D., Chan, J., Tek, H., and Kimia, B. (1998). Symmetry-based indexing of image database. *Journal of Visual Communication and Image Representation (JVCIR)*.

[181] Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[182] Shotton, J., Blake, A., and Cipolla, R. (2005). Contour-based learning for object detection. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[183] Shotton, J., Blake, A., and Cipolla, R. (2008a). Multiscale categorical object recognition using contour fragments. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[184] Shotton, J., Johnson, M., and Cipolla, R. (2008b). Semantic texton forests for image categorization and segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[185] Shridhar, M. and Badreldin, A. (1984). High accuracy character recognition algorithm using fourier and topological descriptors. *Pattern Recognition*.

[186] Siddiqi, K. and Kimia, B. (1996). A shock grammar for recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[187] Sivic, J. and Zisserman, A. (2003). Video google: A text retrieval approach to object matching in videos. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[188] Sonka, M., Hlavac, V., and Boyle, R. (2008). *Image processing, analysis, and machine vision*. Thompson Learning, 3rd edition.

[189] Srinivasan, P. (2011). *Holistic Shape-Based Object Recognition Using Bottom-Up Image Structures*. Phd thesis, University of Pennsylvania.

[190] Srinivasan, P., Zhu, Q., and Shi, J. (2010). Many-to-one contour matching for describing and discriminating object shape. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[191] Stein, A., Hoiem, D., and Hebert, M. (2007). Learning to find object boundaries using motion cues. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[192] Tell, D. and Carlsson, S. (2001). Wide baseline point matching using affine invariants computed from intensity profiles. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[193] Thayananthan, A., Stenger, B., Torr, P., and Cipolla, R. (2003). Shape context and chamfer matching in cluttered scenes. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[194] Toshev, A. (2011). *Shape Representations for Object Recognition*. Phd thesis, University of Pennsylvania.

[195] Toshev, A., Taskar, B., and Daniilidis, K. (2010). Object detection via boundary structure segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[196] Tu, Z., Chen, X., Yuille, A., and Zhu, S. (2005). Image parsing: Unifying segmentation, detection, and recognition. *International Journal of Computer Vision (IJCV)*.

[197] Tu, Z. and Yuille, A. (2004). Shape matching and recognition - using generative models and informative features. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[198] Tu, Z., Zheng, S., and Yuille, A. (1985). Shape Matching and Registration by Data-driven EM. *Computer Vision and Image Understanding (CVIU)*.

[199] Turney, J., Mudge, T., and Volz, R. (1985). Recognizing partially occluded parts. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[200] Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag.

[201] Vedaldi, A., Gulshan, V., Varma, M., and Zisserman, A. (2009). Multiple kernels for object detection. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[202] Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[203] Wang, H. and Oliensis, J. (2008). Shape matching by segmentation averaging. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[204] Weinland, D. and Boyer, E. (2008). Action recognition using exemplar-based embedding. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[205] Winn, J. and Shotton, J. (2006). The layout consistent random field for recognizing and segmenting partially occluded objects. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[206] Wishart, D. (1969). Mode analysis: A generalization of the nearest neighbor which reduces chaining effects. *Numerical Taxonomy*.

[207] Wojek, C. and Schiele, B. (2008). A dynamic conditional random field model for joint labeling of object and scene classes. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[208] Wolfson, H. and Rigoutsos, I. (1997). Geometric hashing: An overview. *Computational Science and Engineering*.

[209] Xu, C., Liu, J., and Tang, X. (2009). 2d shape matching by contour flexibility. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[210] Yang, X., Liu, H., and Latecki, L. (2010a). Contour-based object detection as dominant set computation. In *Proceedings of Asian Conference on Computer Vision (ACCV)*.

[211] Yang, Y., Hallman, S., Ramanan, D., and Fowlkes, C. (2010b). Layered object detection for multi-class segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[212] Yang, Y., Hallman, S., Ramanan, D., and Fowlkes, C. (2011). Layered object models for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*.

[213] Yankov, D. and Keogh, E. (2006). Manifold clustering of shapes. In *Proceedings of International Conference on Data Mining (ICDM)*.

[214] Yarlagadda, P., Monroy, A., and Ommer, B. (2010). Voting by grouping dependent parts. In *Proceedings of European Conference on Computer Vision (ECCV)*.

[215] Yu, S. and Shi, J. (2003). Object-specific figure-ground segregation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[216] Zheng, S., Tu, Z., and Yuille, A. (2007). Detecting object boundaries using low-, mid-, and high-level information. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[217] Zheng, S., Yuille, A., and Tu, Z. (2010). Detecting object boundaries using low-, mid-, and high-level information. *Computer Vision and Image Understanding (CVIU)*.

[218] Zhu, Q. (2011). *Shape Detection by Packing Contours and Regions*. Phd thesis, University of Pennsylvania.

[219] Zhu, Q., Song, G., and Shi, J. (2007). Untangling cycles for contour grouping. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*.

[220] Zhu, Q., Wang, L., Wu, Y., and Shi, J. (2008). Contour context selection for object detection: A set-to-set contour matching approach. In *Proceedings of European Conference on Computer Vision (ECCV)*.